UNIVERSITY OF CALIFORNIA, SAN DIEGO

Using Sparse CCA for Vocabulary Selection

A Thesis submitted in partial satisfaction of the
requirements for the degree Master of Science
in
Computer Science

by

David A. Torres

Committee in charge:

Professor Gert R. G. Lanckriet, Chair
Professor Serge J. Belongie, Co-Chair
Professor Sanjoy Dasgupta

2009

The Thesis of David Anthony Torres is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____
Co-Chair

_____
Chair

University of California, San Diego

2009

TABLE OF CONTENTS

Signature Page iii

Table of Contents iv

List of Figures and Tables vi

Acknowledgements vii

Abstract viii

I 1

1. Introduction 1
2. Related Work 4
3. Acoustic Correlation with Sparse CCA 6
   - A. Canonical Correlation Analysis 7
   - B. Sparse Canonical Component Analysis 9
4. Representing Semantic and Audio Data 14
   - A. Semantic Representation 15
   - B. Audio Representation 17
   - C. CCA Matrices 17
5. Experiments and Results 18
   - A. Selection of Musically Descriptive Tags 18
   - B. Vocabulary Selection for Music Retrieval 20
   - C. Qualitative Discussion 25

6. Discussion 26

References 29

LIST OF FIGURES AND TABLES

Figure I.1 Comparison of vocabulary selection techniques: We compare vocabulary selection using human agreement, acoustic correlation, and a random baseline, as it effects retrieval performance. Acoustic correlation is able to significantly increase the performance of the autotagging system. 21

Table I.1 Shown in bold: The fraction of "higher-quality" tags (CAL500) which comprise a vocabulary as vocabulary size is reduced. 18

Table I.2 Top and bottom 3 tags within semantic categories according to sparse CCA vocabulary selection. 25

ACKNOWLEDGMENTS

I would like to acknowledge Professor Gert Lanckriet for his support as the chair of my committee, and for his encouragement and advice throughout my time at UCSD. I would also like to acknowledge the rest of my committee: Serge Belongie, who infected me with enthusiasm for computer vision and machine learning, and helped set me on the path I am on today; and Sanjoy Dasgupta, whose kindness and clarity of thought have inspired me in my work.

I would also like to acknowledge the founding members of the Computer Audition Laboratory: Professor Doug Turnbull, and Luke Barrington. They have helped me immeasurably in my academic career and without them this thesis would not be possible.

Finally, I would also like to acknowledge Bharath Sriperumbudur, my co-author and friend. Bharath's insightful analytical acumen laid the mathematical bedrock of this thesis.

This thesis, in full, has been submitted for publication of the material as it may appear in IEEE Transactions on Audio, Speech and Language Processing 2009. Torres, David; Sriperumbudur, Bharath; Turnbull, Douglas; Lanckriet, Gert R. G., 2009. The thesis author was the primary investigator and author of this paper.

ABSTRACT OF THE THESIS

Using Sparse CCA for Vocabulary Selection

by

David A. Torres

Master of Science in Computer Science

University of California, San Diego, 2009

Professor Gert R. G. Lanckriet, Chair

Professor Serge J. Belongie, Co-Chair

A content-based *autotagging* system is a computer system that automatically annotates multimedia data such as music, images, and video with tags (semantically-meaningful text-based tokens) based solely on the multimedia content. When developing an autotagging system, three important design decisions are 1) selecting a vocabulary of tags, 2) choosing a feature-based representation of the multimedia content, and 3) picking a supervised learning framework. If we select a tag that cannot be consistently used to annotate multimedia data based on the multimedia content alone (e.g., inconsistent human annotation), or if the feature representation does not encode the information necessary to annotate the multimedia content, then it is unlikely that the supervised

learning framework will be able to successfully annotate novel multimedia content with that tag.

This paper proposes an approach to select a vocabulary of tags based on *sparse canonical component analysis* (sparse CCA). That is, sparse CCA is used to find a set of "acoustically meaningful" tags that are correlated with a chosen feature-based representation of multimedia content. As a result, we find that we are better able to model the selected tags using our supervised autotagging system. In this paper, we specifically focus on music since we are interested in building a content-based music annotation system.

I

I.1 Introduction

A content-based music *autotagging* system is a computer program that can annotate songs with semantically meaningful tags (e.g., "happy", "classic rock", "distorted electric guitar") based on analysis of audio signals [48, 7, 24]. Once songs have been automatically annotated with tags, relevant songs can be retrieved for a given text query using a standard information retrieval framework. We have observed that autotagging performance varies drastically depending on the tag. While there are many reasons for this, (e.g., human subjectivity or polysemy), we focus on two specific reasons in this paper. First, a tag may not be well represented by the audio signal. This may occur if additional contextual information, such as geography or chronology, is needed to accurately use the tag. Second, relevant acoustic information may not be encoded in the acoustic feature representation. For example, if the tag is related to some rhythmic aspect of music (e.g., "waltz") and we extract features which are associated with notions of timbre (e.g., Mel-frequency cepstral coefficients), then the salient information needed to predict the tag may be lost during feature extraction.

In this paper, we explore the problem of *vocabulary selection*, whereby, given an acoustic representation, we identify a set of tags that allows accurate modeling in a supervised learning framework (e.g., support vector machines [51, 24], Gaussian mixture model classifiers [6, 48] and boosted decision stumps [10, 7]). Our primary contribution

is the notion of *acoustic correlation* as an indicator for selecting tags. The idea is to find a set of tags in a vocabulary whose use has a high correlation with the audio feature representation. Such correlation may indicate that an underlying structure exists which is readily modeled by supervised learning methods. While we focus on musical autotagging in the paper, our approach is general and is applicable to any domain where the goal is to annotate multimedia (e.g., images, video, sound) using a vocabulary of tags.

To motivate the problem further, let us outline key findings that we encountered when building our autotagging system that led us to consider vocabulary selection, acoustic correlation and our proposed method of analysis, sparse canonical correlation analysis (sparse CCA).

Thus far, we have collected annotations for music using various methods: For the first iteration of our autotagger, we obtained music tags by text-mining song reviews [45]. In later iterations, we obtained tags by conducting controlled human surveys [47], and we explored the use of a human computation game [49, 53]. In each case, we have been forced to build a vocabulary using ad-hoc methods. For example, the text-mined song reviews resulted in a list of over 1,000 candidate tags, the majority of which were not acoustically relevant. The authors *manually* pruned these tags upon the consensus that a tag was not "acoustically relevant". To collect the survey and game data, we built, *a priori*, a two-level hierarchical vocabulary, first considering a set of high-level semantic categories, (such as Instrumentation, Emotion, Vocal Characteristics, and Genre), and then listing low-level tags for each category (such as "gloomy", "alto saxophone", "falsetto", "hip hop".) It is unsatisfying to manually create a vocabulary that is subject to the biases of its creators. Hence, a key advantage of our vocabulary selection method is that it allows us to select tags based on a quantitative measure.

After assembling a vocabulary and obtaining an annotated corpus of music, i.e., a data set of known song-tag associations, the next steps are to build and evaluate our semantic models. To build the models, we train a statistical model of the acoustic feature vectors associated with each tag. This is done using a subset of the annotated corpus called the training set. Then we evaluate the performance of these models in annotation

and retrieval tasks using songs in the corpus that were not used in training (called the test set).

During evaluation, we found that model performance varied drastically over tags in the vocabulary implying that some tags were being modeled well, and others quite poorly (i.e., not much better than random guessing). One reason for this variation is that humans themselves inconsistently or inaccurately annotate music with tags. For example, two people may disagree whether a song is "sad". Another person may incorrectly identify an "alto saxophone" as a "clarinet". These kinds of inconsistencies will always be problematic for autotagging systems; however, in this paper, we focus on another more tractable problem that degrades autotagging performance.

The problem occurs when a tag is not well represented by the acoustic features extracted from the audio signal. If this is the case, then it is likely that the audio representation lacks the expressive power necessary to encode the semantics of this tag. This translates to poor performance of the statistical models. This issue is the primary motivation behind our vocabulary selection technique: To create an effective autotagging system, we must recognize which tags are candidates for successful modeling.

To address this problem, we propose the notion of acoustic correlation. We consider a set of tags to be acoustically correlated with an acoustic representation if there exists a strong correlation (in the mathematical sense) between the tags' use and the extracted audio features. If a strong correlation can be found, then the tag-audio relationship may be sufficiently salient to allow successful modeling of the tags. Consider this simple but motivating example for a single tag: It is reasonable to expect that loudness, a property of the audio content, is correlated with the tag "hard rock", therefore we should be able to use loudness to help distinguish between "hard rock" and non-"hard rock" music.

To measure acoustic correlation we propose the use of an unsupervised statistical method based on *canonical correlation analysis* (CCA). CCA is a method of exploring cross-correlations between different representations of data. Within the CCA framework it is assumed that these representations "share joint information that is reflected in

correlations between them [4].” Similar to principal component analysis (PCA), where we find informative basis vectors that maximize a data’s projected variance, CCA finds *pairs* of basis vectors that maximize the data’s projected cross-correlation.[1]

Given music data represented in both a semantic feature space (audio tags) and an acoustic feature space, the output of CCA, as it is used in this paper, is a weighted combination of tags where the magnitude of each weight is an indication of how strongly the corresponding tag contributes to the acoustic correlation. An immediate vocabulary selection technique presents itself: Keep those tags with a high absolute weight, and remove those tags with weight close to zero. While this method works in many cases, it has been shown that this type of threshold-based variable selection may not always give the best results [59, 17]. If the goal is to select a subset of tags, then it is best to encode that information into the algorithm itself. This is done by explicitly seeking solutions with zero weights for uninformative tags. This type of analysis is known as *sparse analysis* and leads to the method presented in this paper, sparse *canonical correlation analysis* (sparse CCA).

The rest of this paper is organized as follows. Section I.2 discusses related work in the realm of vocabulary selection, CCA applications and sparse methods. Section I.3 introduces the CCA method and introduces the concept of sparsity. This section also derives our sparse CCA algorithm. Section I.4 describes semantic and audio representations of the data sets used in this paper. Section I.5 explains our experiments and discusses our results. Finally Section I.6 concludes.

I.2 Related Work

The explosion of digital music on the Internet has led to both commercial (e.g., Pandora [2], Last.FM [3], AMG [4], and Apple iTunes [5]) and academic interest (e.g., [48, 7,

[1] CCA is a direct generalization of PCA for multiple feature spaces.
[2] http://pandora.com
[3] http://www.last.fm
[4] http://allmusic.com
[5] http://www.apple.com/itunes/

36, 24, 20, 27, 55]) in music search technology. Recently, eleven autotagging systems were compared head-to-head in the "Audio Tag Classification" task of the 2008 Music Information Retrieval Evaluation eXchange (MIREX) [5]. Due to multiple evaluation metrics and lack of statistical significance in the results, there was no overall winning system, but our system was the top performing system for a number of the evaluation metrics [48]. Our autotagging algorithm, which we use in our experiments section and is briefly described in this paper, uses a generative statistical approach that learns a Gaussian mixture model (GMM) distribution over an audio feature space for each tag in the vocabulary.

Mandel and Ellis proposed another top-performing approach in which they learn a binary support vector machine (SVM) for each tag in the vocabulary [24]. They used Platt scaling [30] to convert SVM decision function scores to probabilities so that tag relevance could be compared across multiple SVMs. Eck et. al. [7] also used a discriminative approach by learning a boosted decision stump classifier for each tag. Finally, Sordo et. al. [36] presented a non-parametric approach that used a content-based measure of music similarity to propagate tags from annotated songs to similar songs that had not been annotated. It should be noted that autotagging is an extension of content-based music classification by genre [50, 26], emotion [21], and instrumentation [9].

Other researchers have explored tag selection methods for a music annotation task [27, 32, 20, 55]. Roy and Pachet analyzed labeled music and, using a SVM, tested the hypothesis that song popularity can be predicted from human features or acoustic features [27]. Whitman and Ellis [55] used a technique based on a SVM to analyze album reviews jointly with audio content. They created a system than can prune subjective words and sentences from the reviews. Other work has focused on exploring semantic relationships between tags without considering acoustic information [32, 20].

In all cases, the authors explore the quality of their autotagging systems in *a posteriori* fashion. That is, they evaluate their ability to annotate songs after annotation is performed by their supervised learning system. To our knowledge, our work is the first proposed *a priori* technique in which we select a vocabulary of tags *before* attempting

to autotag music with a supervised learning method.

Our vocabulary selection approach is based on the use of our novel formulation of sparse CCA [44, 43]. Others have converged on CCA to perform semantic analysis of different types of media. Hardoon et al. used kernel CCA to create semantic models of images and accompanying text [13]. In addition, researchers have made use of kernel CCA for cross-language document retrieval [52, 22]. These applications use kernel CCA directly to annotate and retrieve data, which is a markedly different approach than ours, in which we emphasize using sparse CCA as a *pre-processing step* before performing further modeling.

Sparse CCA research (as compared to regular CCA or kernel CCA [4, 12]) is a Newer line of research that has followed from the sparse PCA literature [17, 59, 3, 38]. We have developed an algorithm that finds sparse solutions to generalized eigenvalue problems which has direct applications to sparse CCA [38, 39]. The double-barreled LASSO [11], a close cousin of the LASSO technique [41], solves a sparse CCA problem by framing it as a convex least squares problem. Finally, Kidron et. al, used sparse CCA to find pixels in video that were correlated to an accompanying audio track [18]. They solved a sparse CCA problem by constraining regular CCA solutions with an ℓ_1 -norm.

I.3 Acoustic Correlation with Sparse CCA

This section describes the sparse CCA algorithm. First we introduce CCA and show that it reduces to solving a generalized eigenvalue problem of the form $\mathbf{Aw} = \lambda\mathbf{Bw}$, (where $\mathbf{A}$ is symmetric, $\mathbf{B}$ is symmetric positive definite, and $\mathbf{w} \in \mathbb{R}^n$). Next, we introduce the notion of sparsity and show how we impose sparsity constraints on the generalized eigenvalue problem. A direct solution to a sparsely constrained eigenvalue problem is intractable, so we detail an approximating algorithm that we have developed to solve it.

I.3.A Canonical Correlation Analysis

Canonical Correlation Analysis (CCA) is a method by which we find correlations between different representations of some underlying phenomenon. For example, consider some phenomenon that gives rise to a pair of observations, which we encode in the observation vectors $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$, where n does not (necessarily) equal m. In this paper we assume that the underlying phenomenon is an instance of a song which leads to two types of observation vectors: The first is an annotation vector of the song, a collection of tags that is used by humans to describe it. The second observation vector is a description of the audio content, such as statistics derived from the song's spectral data.

Given a collection of annotated songs, i.e., annotation vectors and audio feature vectors, it is interesting to consider whether there exist elements in the observation vectors that are cross-correlated. This knowledge is useful because correlations between sets of tags and audio content indicate an underlying linear structure that can be modeled, analyzed and potentially exploited. In this paper we deal specifically with the case of semantic music analysis, however, the idea applies to any situation where one has intrinsically heterogeneous types of data describing the same underlying phenomenon.

CCA is a tool to discover these types of cross-correlations. Formally, consider a collection of pairs of observation vectors where each pair is contained in corresponding rows of the matrices $\mathbf{X} \in \mathbb{R}^{p \times n}$ and $\mathbf{Y} \in \mathbb{R}^{p \times m}$ respectively, where p is the number of observations. The CCA problem seeks to find a pair of basis vectors, one in each dataspace, $\mathbf{w_x} \in \mathbb{R}^n$ and $\mathbf{w_y} \in \mathbb{R}^m$, such that the correlation of the projected observations is maximized. Mathematically,

$$\arg\max \left\{ \text{Corr}(\mathbf{X}\mathbf{w_x}, \mathbf{Y}\mathbf{w_y}) := \frac{\mathbf{w_x}^T \mathbf{\Sigma_{xy}} \mathbf{w_y}}{\sqrt{\mathbf{w_x}^T \mathbf{\Sigma_{xx}} \mathbf{w_x}} \sqrt{\mathbf{w_y}^T \mathbf{\Sigma_{yy}} \mathbf{w_y}}} \right\},$$

where $\text{Corr}(\mathbf{X}\mathbf{w_x}, \mathbf{Y}\mathbf{w_y})$ represents the correlation between $\mathbf{X}\mathbf{w_x}$ and $\mathbf{Y}\mathbf{w_y}$. $\mathbf{\Sigma_{xy}}$ represents the cross-covariance matrix associated with $\mathbf{X}$ and $\mathbf{Y}$ while $\mathbf{\Sigma_{xx}}$ and $\mathbf{\Sigma_{yy}}$ represent the covariance matrices associated with $\mathbf{X}$ and $\mathbf{Y}$ respectively. This problem is

equivalent to

$$\arg\max \left\{ \mathbf{w}_\mathbf{x}^T \mathbf{\Sigma}_{\mathbf{xy}} \mathbf{w}_\mathbf{y} \ : \ \mathbf{w}_\mathbf{x}^T \mathbf{\Sigma}_{\mathbf{xx}} \mathbf{w}_\mathbf{x} = 1, \ \mathbf{w}_\mathbf{y}^T \mathbf{\Sigma}_{\mathbf{yy}} \mathbf{w}_\mathbf{y} = 1 \right\}, \tag{I.1}$$

which can be reposed as a generalized eigenvalue problem as follows. Taking the Lagrangian of Eq. (I.1), which is given by

$$\mathcal{L}(\mathbf{w}_\mathbf{x}, \mathbf{w}_\mathbf{y}, \lambda_\mathbf{X}, \lambda_\mathbf{Y}) = \mathbf{w}_\mathbf{x}^T \mathbf{\Sigma}_{\mathbf{xy}} \mathbf{w}_\mathbf{y} - \frac{\lambda_\mathbf{X}}{2} \mathbf{w}_\mathbf{x}^T \mathbf{\Sigma}_{\mathbf{xx}} \mathbf{w}_\mathbf{x} - \frac{\lambda_\mathbf{Y}}{2} \mathbf{w}_\mathbf{y}^T \mathbf{\Sigma}_{\mathbf{yy}} \mathbf{w}_\mathbf{y}, \tag{I.2}$$

we differentiate $\mathcal{L}$ with respect to $\mathbf{w}_\mathbf{x}$ and $\mathbf{w}_\mathbf{y}$. Setting these derivatives to zero yields the following Karush-Kuhn-Tucker conditions which are the necessary conditions to be satisfied at the optimum. Here $\lambda_\mathbf{X}, \lambda_\mathbf{Y} \in \mathbb{R}$,

$$\mathbf{\Sigma}_{\mathbf{xy}} \mathbf{w}_\mathbf{y} = \lambda_\mathbf{X} \mathbf{\Sigma}_{\mathbf{xx}} \mathbf{w}_\mathbf{x}, \tag{I.3}$$

$$\mathbf{\Sigma}_{\mathbf{yx}} \mathbf{w}_\mathbf{x} = \lambda_\mathbf{Y} \mathbf{\Sigma}_{\mathbf{yy}} \mathbf{w}_\mathbf{y}. \tag{I.4}$$

Pre-multiplying Eq. (I.3) by $\mathbf{w}_\mathbf{x}$ and Eq. (I.4) by $\mathbf{w}_\mathbf{y}$, we have,

$$\lambda_\mathbf{X} \mathbf{w}_\mathbf{x}^T \mathbf{\Sigma}_{\mathbf{xx}} \mathbf{w}_\mathbf{x} = \mathbf{w}_\mathbf{x}^T \mathbf{\Sigma}_{\mathbf{xy}} \mathbf{w}_\mathbf{y} = \mathbf{w}_\mathbf{y}^T \mathbf{\Sigma}_{\mathbf{yx}} \mathbf{w}_\mathbf{x} = \lambda_\mathbf{Y} \mathbf{w}_\mathbf{y}^T \mathbf{\Sigma}_{\mathbf{yy}} \mathbf{w}_\mathbf{y}, \tag{I.5}$$

and since $\mathbf{w}_\mathbf{x}^T \mathbf{\Sigma}_{\mathbf{xx}} \mathbf{w}_\mathbf{x} = \mathbf{w}_\mathbf{y}^T \mathbf{\Sigma}_{\mathbf{yy}} \mathbf{w}_\mathbf{y} = 1$, we have $\lambda_\mathbf{X} = \lambda_\mathbf{Y} =: \lambda$. This implies that Eqs. (I.3-I.4) can be written as,

$$\mathbf{\Sigma}_{\mathbf{xy}} \mathbf{w}_\mathbf{y} = \lambda \mathbf{\Sigma}_{\mathbf{xx}} \mathbf{w}_\mathbf{x}, \tag{I.6}$$

$$\mathbf{\Sigma}_{\mathbf{yx}} \mathbf{w}_\mathbf{x} = \lambda \mathbf{\Sigma}_{\mathbf{yy}} \mathbf{w}_\mathbf{y}, \tag{I.7}$$

which is equivalent to the generalized maximum eigenvalue problem, $\mathbf{A}\mathbf{w} = \lambda \mathbf{B}\mathbf{w}$, where

$$\mathbf{A} = \begin{pmatrix} \mathbf{0} & \mathbf{\Sigma}_{\mathbf{xy}} \\ \mathbf{\Sigma}_{\mathbf{yx}} & \mathbf{0} \end{pmatrix}, \mathbf{B} = \begin{pmatrix} \mathbf{\Sigma}_{\mathbf{xx}} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_{\mathbf{yy}} \end{pmatrix} \text{ and } \mathbf{w} = \begin{pmatrix} \mathbf{w}_\mathbf{x} \\ \mathbf{w}_\mathbf{y} \end{pmatrix}.$$

I.3.B Sparse Canonical Component Analysis

The solution vector, $\mathbf{w}$, of the generalized eigenvalue problem is generally *non-sparse*. That is, most of its elements are non-zero. However, in many applications it is necessary to limit the number of non-zero elements in $\mathbf{w}$, or, in other words, to impose sparsity on $\mathbf{w}$. Sparsity can aid the interpretability of experimental results. In many cases, the elements of $\mathbf{w}$ correspond to experimental variables. By imposing sparsity on $\mathbf{w}$ and keeping those variables that correspond to non-zero elements, we obtain a subset of the variables that are the most informative or the most relevant to an experiment, thereby performing feature selection.

For example, sparsity is often desired in biological applications [14, 59, 57]. Some of these applications consist of the analysis of gene expression data in which each observation vector contains thousands of input variables which correspond to the expression levels of individual genes. These can be thought of, roughly, as real valued weights that express how well a gene responded in some experiment. Employing sparse statistical methods such as sparse principal component analysis or sparse least squares regression allows researchers to find genes (input variables in $\mathbf{w}$) that are the most critical to explaining an experimental model. The benefit of this is better interpretability of the experiment. For example, sparsity can be used to discover which genes play a roll in diseases, or to create a simplified experimental setup, since the experimenter now knows which genes to target in future experiments.

Sparsity can also be used to compress information. Consider the problem of efficiently representing a signal $\mathbf{v} \in \mathbb{R}^n$ from a set of basis functions contained in the columns of the matrix Φ. Mathematically, we seek a weight vector $\mathbf{w}$ that satisfies $\Phi\mathbf{w} = \mathbf{v}$. Due to efficiency requirements the number of basis functions may be prohibitively large, making the encoding and decoding of the signal processor intensive. In such cases we would rather seek a *sparse* $\mathbf{w}$ such that $\Phi\mathbf{w} \approx \mathbf{v}$, hence providing a means of doing lossy compression [42, 56].

Mathematically, we impose a sparsity constraint on any optimization problem

in the following way. Assume that the optimization problem is over the vector $\mathbf{w}$. We seek a solution to the problem with the additional restriction that $\text{card}(\mathbf{w}) \leq k$, where $\text{card}(\mathbf{w})$ represents the number of non-zero elements of $\mathbf{w}$. (In this paper, we use the notation $\|\mathbf{w}\|_0$ to represent $\text{card}(\mathbf{w})$). This sparsity constraint is problematic in that its combinatorial nature renders the problem NP hard. Usually an ℓ_1 -norm relaxation to $\text{card}(\mathbf{w})$ is used to make the problem convex, which allows polynomial time algorithms to solve it [1]. However, in our case this relaxation is not helpful since it can be seen in Eq. (I.9) that adding this constraint to the generalized eigenvalue problem does not result in a convex program. Therefore, one can use better approximations to $\text{card}(\mathbf{w})$ than the ℓ_1 -norm. Different approximations to $\text{card}(\mathbf{w})$ have been proposed in literature. One proposition [54] is to replace $\text{card}(\mathbf{w})$ by $\sum_{i=1}^{n} \log(\epsilon + |w_i|)$, where $\epsilon > 0$, another [2] uses $\sum_{i=1}^{n}(1 - e^{-\alpha|w_i|})$ for $\alpha > 0$. These approximations were used in the context of feature selection using support vector machines. In this paper, we use an approximation similar to that of [54],

$$\|\mathbf{w}\|_0 \approx \sum_{i=1}^{n} \frac{\log(1 + \epsilon^{-1}|w_i|)}{\log(1 + \epsilon^{-1})}, \tag{I.8}$$

for some $\epsilon > 0$, and solve the resulting problem as a continuous optimization problem. This approximation is related to sparse priors, which are studied in Bayesian inference. Specifically, this approximation can be interpreted as defining a limiting Student's t-distribution prior over $\mathbf{w}$, an improper prior given by $p(\mathbf{w}) \propto \prod_{i=1}^{n} \frac{1}{|w_i|+\varepsilon}$ and computing its negative log-likelihood [54]. Others have shown that this choice of prior leads to a sparse representation and have demonstrated its validity for sparse kernel expansions in a Bayesian framework [42]. Finally, it can be shown that the approximation improves monotonically and approaches $\text{card}(\mathbf{w})$ as ϵ approaches zero [39] [54].

In what follows, we show how to incorporate this approximation into the generalized eigenvalue problem of the form $\mathbf{A}\mathbf{w} = \lambda\mathbf{B}\mathbf{w}$, which leads to the sparse CCA algorithm. First, let us consider the variational formulation of the generalized eigenvalue problem given by

$$\max\{\mathbf{w}^T\mathbf{A}\mathbf{w} \,:\, \mathbf{w}^T\mathbf{B}\mathbf{w} = 1\}, \tag{I.9}$$

where $\mathbf{A}$ is a symmetric matrix and $\mathbf{B}$ is a positive definite matrix. The associated cardinality constrained version of the problem is given by

$$\begin{aligned} \max_{\mathbf{w}} \quad & \mathbf{w}^T\mathbf{A}\mathbf{w} \\ s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1 \\ & ||\mathbf{w}||_0 \leq k, \end{aligned} \tag{I.10}$$

for some $1 \leq k \leq n$. By absorbing the cardinality constraint into the objective, the problem is equivalent to

$$\begin{aligned} \max_{\mathbf{w}} \quad & \mathbf{w}^T\mathbf{A}\mathbf{w} - \rho\|\mathbf{w}\|_0 \\ s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1, \end{aligned} \tag{I.11}$$

where $\rho \geq 0$. Now, we solve the following approximate problem wherein $\|\mathbf{w}\|_0$ is replaced by its approximation,

$$\begin{aligned} \max_{\mathbf{w}} \quad & \mathbf{w}^T\mathbf{A}\mathbf{w} - \frac{\rho}{\log(1+\epsilon^{-1})}\sum_{i=1}^{n}\log(1+\epsilon^{-1}|w_i|) \\ s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1. \end{aligned} \tag{I.12}$$

Let $Q(\mathbf{w}) = \mathbf{w}^T\mathbf{A}\mathbf{w} - \rho\|\mathbf{w}\|_0$, $Q_\epsilon(\mathbf{w}) = \mathbf{w}^T\mathbf{A}\mathbf{w} - \frac{\rho}{\log(1+\epsilon^{-1})}\sum_{i=1}^{n}\log(1+\epsilon^{-1}|w_i|)$ and $\Omega = \{\mathbf{w} : \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1\}$. Let $\widehat{\mathbf{w}}$ and $\mathbf{w}_\epsilon$ be the maximizers of Eq. (I.11) and Eq. (I.12) respectively for fixed ρ and ϵ. Then, we can show that $|Q_\epsilon(\mathbf{w}_\epsilon) - Q(\widehat{\mathbf{w}})| \rightarrow 0$ as $\epsilon \rightarrow 0$, which means that, asymptotically, the optimal values of Eq. (I.11) and Eq. (I.12) are the same [39]. Choosing a small value for ϵ gives a solution $\mathbf{w}_\epsilon$ such that $|Q(\widehat{\mathbf{w}}) - Q_\epsilon(\mathbf{w}_\epsilon)|$ is small. Therefore, one can think of $\mathbf{w}_\epsilon$ as an approximate solution to the problem in Eq. (I.11). Note that ρ has to be fixed before solving Eq. (I.12) to achieve the desired sparsity. For a fixed ϵ, we let $\tilde{\rho} = \frac{\rho}{\log(1+\epsilon^{-1})}$ and fix $\tilde{\rho}$.

The approximate sparse eigenvalue problem is now given by

$$
\begin{aligned}
-\min_{\mathbf{w}} \quad & -\mathbf{w}^T\mathbf{A}\mathbf{w} + \tilde{\rho}\sum_{i=1}^{n}\log\left(|w_i|+\epsilon\right) \\
s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1,
\end{aligned}
\tag{I.13}
$$

which is equivalent to

$$
\begin{aligned}
-\min_{\mathbf{w},\mathbf{y}} \quad & -\mathbf{w}^T\mathbf{A}\mathbf{w} + \tilde{\rho}\sum_{i=1}^{n}\log\left(y_i+\epsilon\right) \\
s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1 \\
& -y_i \leq w_i \leq y_i.
\end{aligned}
\tag{I.14}
$$

Since $\mathbf{A}$ may be indefinite, the problem can be equivalently written as

$$
\begin{aligned}
-\min_{\mathbf{w},\mathbf{y}} \quad & \tau\|\mathbf{w}\|_2^2 \\
& -\left(\mathbf{w}^T(\mathbf{A}+\tau\mathbf{I})\mathbf{w} - \tilde{\rho}\sum_{i=1}^{n}\log\left(y_i+\epsilon\right)\right) \qquad \text{(I.15)} \\
s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1 \\
& -y_i \leq w_i \leq y_i, \qquad \text{(I.16)}
\end{aligned}
$$

where $\tau \geq \max(0, -\lambda_{min}(\mathbf{A}))$ so that $\mathbf{A}+\tau\mathbf{I}$ is positive semidefinite. $\lambda_{min}(\mathbf{A})$ represents the minimum eigenvalue associated with $\mathbf{A}$. The reason for rewriting Eq. (I.14) as Eq. (I.15) is that the objective function in Eq. (I.15) becomes a difference of convex functions (d.c.) and therefore the problem in Eq. (I.15) is a d.c. program.[6] D.c. programs can be solved using DCA [40, 15] or CCCP [58] (which are iterative methods), where the idea is to linearize the concave part of the objective function, i.e., $-\mathbf{w}^T(\mathbf{A}+\tau\mathbf{I})\mathbf{w} + \tilde{\rho}\sum_{i=1}^{n}\log\left(y_i+\epsilon\right)$, around some point that lies in the constraint

[6] Let $\mathscr{C}$ be a convex set of $\mathbb{R}^n$. A real valued function $f : \mathscr{C} \to \mathbb{R}$ is called a d.c. on $\mathscr{C}$, if there exist two *convex* functions $g, h : \mathscr{C} \to \mathbb{R}$ such that f can be expressed in the form $f(\mathbf{x}) = g(\mathbf{x}) - h(\mathbf{x})$, $\mathbf{x} \in \mathscr{C}$. Optimization problems of the form $\min_{\mathbf{x}}\{f_0(\mathbf{x}) : \mathbf{x} \in \mathscr{C}, f_i(\mathbf{x}) \leq 0, i = 1, \ldots, m\}$, where $f_i = g_i - h_i$, $i = 0, \ldots, m$, are d.c. functions are called *d.c. programs*.

set so that the resulting program is convex. Formally, for an iteration $l+1$, we have

$$\begin{aligned}(\mathbf{w}^{(l+1)}, \mathbf{y}^{(l+1)}) \in \arg\min_{\mathbf{w},\mathbf{y}} \quad & \tau\|\mathbf{w}\|_2^2 \\ & -2\mathbf{w}^T(\mathbf{A}+\tau\mathbf{I})\mathbf{w}^{(l)} \\ & +\tilde{\rho}\sum_{i=1}^{n}\frac{y_i}{y_i^{(l)}+\epsilon} \\ s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1 \\ & -y_i \leq w_i \leq y_i, \end{aligned} \tag{I.17}$$

which is equivalent to

$$\begin{aligned}\mathbf{w}^{(l+1)} \in \arg\min_{\mathbf{w}} \quad & \tau\|\mathbf{w}\|_2^2 - 2\mathbf{w}^T(\mathbf{A}+\tau\mathbf{I})\mathbf{w}^{(l)} \\ & +\tilde{\rho}\sum_{i=1}^{n}\frac{|w_i|}{|w_i^{(l)}|+\epsilon} \\ s.t. \quad & \mathbf{w}^T\mathbf{B}\mathbf{w} \leq 1, \end{aligned} \tag{I.18}$$

which is a sequence of convex programs, and specifically, quadratically constrained quadratic programs (QCQPs) [1]. The proposed algorithm starts at some $\mathbf{w}^{(0)} \in \Omega$, computes $\mathbf{w}^{(l+1)}$ by Eq. (I.18) and iterates until $\mathbf{w}^{(l+1)} = \mathbf{w}^{(l)}$, which is the point of convergence. See our previous publication [37] for details related to the convergence of the algorithm.

In the case of sparse CCA we have that $\mathbf{A} = \begin{pmatrix} 0 & \mathbf{\Sigma_{xy}} \\ \mathbf{\Sigma_{yx}} & 0 \end{pmatrix}$, $\mathbf{B} = \begin{pmatrix} \mathbf{\Sigma_{xx}} & 0 \\ 0 & \mathbf{\Sigma_{yy}} \end{pmatrix}$, and $\mathbf{w} = \begin{pmatrix} \mathbf{w_x} \\ \mathbf{w_y} \end{pmatrix}$. Rewriting Eq. (I.18) in terms of these variables we have

$$
\begin{aligned}
(\mathbf{w_x}^{(l+1)}, \mathbf{w_y}^{(l+1)}) \in \arg\min_{\mathbf{w_x},\mathbf{w_y}} \quad & \tau\|\mathbf{w_x}\|_2^2 + \tau\|\mathbf{w_y}\|_2^2 \\
& -2\mathbf{w_x}^T(\mathbf{\Sigma_{xy}}\mathbf{w_y}^{(l)} + \tau\mathbf{w_x}^{(l)}) \\
& -2\mathbf{w_y}^T(\mathbf{\Sigma_{yx}}\mathbf{w_x}^{(l)} + \tau\mathbf{w_y}^{(l)}) \\
& +\tilde{\rho}_\mathbf{x}\sum_i \frac{|\mathbf{w_x}|_i}{|\mathbf{w_x}^{(l)}|_i + \epsilon} \\
& +\tilde{\rho}_\mathbf{y}\sum_i \frac{|\mathbf{w_y}|_i}{|\mathbf{w_y}^{(l)}|_i + \epsilon} \\
\text{s.t.} \quad & \mathbf{w_x}^T\mathbf{\Sigma_{xx}}\mathbf{w_x} \leq 1 \\
& \mathbf{w_y}^T\mathbf{\Sigma_{yy}}\mathbf{w_y} \leq 1,
\end{aligned} \tag{I.19}
$$

where we can impose different sparsity constraints on $\mathbf{w_x}$ and $\mathbf{w_y}$ by tuning the parameters $\tilde{\rho}_\mathbf{x}$ and $\tilde{\rho}_\mathbf{y}$ independently. In our experiments, for example, we set $\tilde{\rho}_\mathbf{x}$ and $\tilde{\rho}_\mathbf{y}$ independently since we are concerned with imposing sparsity on the semantic space but not the audio space.

I.4 Representing Semantic and Audio Data

This section describes the audio and semantic representations used in our experiments. In one of our experiments, we demonstrate how our vocabulary selection method is used to select high quality music tags over low quality ones. Therefore, we make use of two sources of annotated music. The first, high quality source, is the CAL500 [47] data set, which was obtained through a large human survey. The second, lower quality source, is the Web2131 [45] data set, a collection of song annotations obtained by text mining professionally-edited song reviews. We also describe the audio representation used to characterize songs. Finally we describe how this data is used in the sparse CCA algorithm.

I.4.A Semantic Representation

Each song is associated with a set of tags describing the song's semantic content. The tags are selected from a vocabulary $\mathcal{V}$ of fixed size $|\mathcal{V}|$. Internally, we represent these tag associations as *annotation vectors*. Each annotation vector $\mathbf{y}$ contains $|\mathcal{V}|$ elements. Element y_i is positive if and only if tag i has been associated with the song. The value of y_i is a positive real number between 0 and 1 which represents the strength of association of a tag. In our experiments we obtained tags by two methods. We generated one vocabulary by text-mining on-line song reviews, and another by conducting a controlled survey of undergraduate students. We will describe these two data sets in detail in the next two subsections.

Web2131

An effective way to collect semantic information about music is to analyze text documents that are downloaded from the Internet. By analyzing these documents one can collect information related to songs, albums and artists. In this paper we use the Web2131 data set, which is a collection of song annotations that were obtained for 2131 songs. The annotations were extracted from professionally-edited song reviews that were downloaded from All Music Guide (AMG)[7], a popular music-oriented website. We removed common stop words from each review and obtained a large set of the most commonly used tags (unigrams and bigrams). From this set, we hand selected a vocabulary of 317 musically informative tags, meaning that the tags can be used (by the authors' best judgments) to describe something about the audio content of a song. For example, we retain tags such as "blues", "Jew's harp" and "intense" while we remove tags like "across", "catastrophic", and "difference".

[7] http://www.allmusic.com

CAL500

The CAL500 data set is a carefully annotated music corpus of 500 western popular songs by 500 unique artists. We paid 66 undergraduate music students to annotate our music corpus with semantic tags. These tags were created specifically for a music annotation task. The tags consisted of 135 acoustically-relevant concepts spanning six semantic categories: instrumentation, vocal characteristics, genre, emotional words, usage terms, and acoustically descriptive words. Specifically, 29 instruments were annotated as present in the song or not; 22 vocal characteristics were annotated as relevant to the singer or not; 36 genres, a subset of the Codaich genre list [25], were annotated as relevant to the song or not; 18 emotions, found by Skowronek et al. [34] to be both important and easy to identify, were rated on a scale from one to three (e.g. ”not happy”, ”neutral”, ”happy”); 15 song concepts describing the acoustic qualities of the song, artist and recording (e.g., tempo, energy, sound quality) were rated on a scale from one to five; and 15 usage terms from [16], (e.g., “I would listen to this song while *driving, sleeping, etc.*”) were rated as relevant or not. Each song was annotated by a minimum of 3 individuals.

The 135 concepts are converted to a 174-tag vocabulary by mapping bi-polar concepts to multiple tags. For example ‘Energy Level’ maps to ‘low energy’ and ‘high energy’. Then we prune all tags that are represented in five or fewer songs to remove under-represented tags. Lastly, for each song, we construct a real-valued 174-dimensional annotation vector by averaging the label frequencies over individual annotators. Further details concerning the CAL500 data set can be found in our previous publication [47]. In short, each element in an annotation vector is a real-valued scalar which can be thought of as indicating the strength of association a song has to a given tag.

I.4.B Audio Representation

The audio content of a song is represented as a *bag-of-feature-vectors*, an unordered set of vectors $\{\mathbf{x}_1, \mathbf{x}_2 \ldots \mathbf{x}_t\}$. Each vector consists of dynamic Mel-frequency cepstral coefficients (dMFCC) taken from half-overlapping, medium-time segments of audio (every $\sim$743 ms), as explained below [26].

Mel-frequency cepstral coefficients (MFCC) describe the spectral shape of a short-time audio frame and are popular in speech recognition and music classification applications (e.g., [31, 23, 35]). We calculate 13 MFCC coefficients for each short-time ($\sim$23 msec) frame of audio. For each of the 13 MFCCs, we take a discrete Fourier transform (DFT) over a time series of 64 frames, normalize by the DC value (to remove the effect of volume) and summarize the resulting spectrum by integrating across 4 modulation frequency bands: (un-normalized) DC, 1-2Hz, 3-15Hz and 20-43Hz. Thus, we create a 52-dimensional feature vector (4 features for each of the 13 MFCCs) for every $3/4$ second segment of audio. For a five minute song, this results in about 400 52-dimensional feature vectors.

I.4.C CCA Matrices

To perform sparse CCA, we must pack the audio feature vectors and corresponding annotation vectors, for all songs, into two respective matrices, $\mathbf{X}$ and $\mathbf{Y}$. Since we have multiple audio vectors per song, and only one annotation vector per song, we replicate the annotation vector for each audio vector. In mathematical notation, if a song's audio content is represented as the bag-of-feature-vectors above and the song annotation is $\mathbf{y}$, then the pairs of audio and annotation vectors can be written as $\{(\mathbf{x}_1, \mathbf{y}), (\mathbf{x}_2, \mathbf{y}), \ldots, (\mathbf{x}_t, \mathbf{y})\}$. To create the matrices used in sparse CCA, we simply stack the corresponding audio and annotation vectors for *all* songs into the matrices $\mathbf{X}$ and $\mathbf{Y}$. So, for example, if our music data set consisted of 100 5-minute songs, then $\mathbf{X}$ would be $40,000 \times 52$ and $\mathbf{Y}$ would be $40,000 \times |\mathcal{V}|$. (Remember, each 5 minute song $\approxeq$ 400 vectors).

Table I.1: Shown in bold: The fraction of "higher-quality" tags (CAL500) which comprise a vocabulary as vocabulary size is reduced.

vocab. size	488	249	203	149	103	50
# CAL500 tags	173	118	101	85	65	39
# Web2131 tags	315	131	102	64	38	11
%Cal500	**.36**	**.48**	**.50**	**.58**	**.64**	**.78**

The size of these matrices seems prohibitively large, however, keep in mind that sparse CCA deals with the inner product matrices, $\mathbf{X}^T\mathbf{X}$, $\mathbf{Y}^T\mathbf{Y}$ and $\mathbf{X}^T\mathbf{Y}$, which are of sizes (52×52), $(|\mathcal{V}| \times |\mathcal{V}|)$ and $(52 \times |\mathcal{V}|)$ respectively. In practice one can compute these matrices without exhausting system memory.

I.5 Experiments and Results

In this section, we describe two quantitative experiments that illustrate how sparse CCA can be used to perform vocabulary selection. We also provide a short qualitative discussion of some of the results. While we focus on music, this technique can be used in any context (movies, sound effects, images) where we have one (or more) content-based representations and one (or more) semantic representations of each multimedia object.

I.5.A Selection of Musically Descriptive Tags

In our first experiment, we show how sparse CCA can be used to select a set of acoustically relevant tags that are well suited to a music annotation task. Humans can consistently annotate previously unknown songs with certain tags, therefore, it is likely that the underlying information relating the tags and songs must be encoded in the audio content. We propose to look for such information in the form of acoustic correlation between the semantic and acoustic representations of songs. We propose that a set of tags which can explain a significant amount of this acoustic correlation is "acoustically

meaningful." Sparse CCA is the tool that we use to find such a set of tags.[8]

As a proof of concept, we constructed an experiment using both the Web2131 tags and the CAL500 tags. In an informal study, we found the Web2131 tags to be of lower quality than the CAL500 tags, in that there seem to be far fewer tags in Web2131 data that are descriptive of their related songs. To conduct this informal study, we ran a test in which we showed subjects a song annotation of ten tags taken from either the ground truth, Web2131 annotations or ten random tags selected from the vocabulary. Then we asked the subjects to select which group of tags was the most relevant for a given song. We found that Web2131 annotations were not much better than selecting tags randomly from the vocabulary. A similar test with CAL500 tags showed that those annotations were significantly better. Probing further, we used our autotagging system (which we will describe in the next section) to model the music-tag relationships within both Web2131 and CAL500 data sets. We found that tags from the Web2131 vocabulary resulted in poorer performance when evaluating the system. The lower quality of the Web2131 tags makes sense on an intuitive level: The Web2131 tags were culled from music reviews. When these reviews were written, their authors were not choosing individual words based on independent notions of semantic appropriateness. Compare this to the CAL500 tags in which human subjects were explicitly asked to select tags in a music annotation task.

Based on this evidence we assume that the CAL500 data set contains more acoustically descriptive tags than Web2131. Assuming that some proportion of these tags are well encoded in the acoustic feature representation, we expect that the CAL500 vocabulary should contain more acoustically correlated tags than the Web2131 vocabulary. Therefore, if we perform sparse CCA vocabulary selection on a set of songs with both CAL500 and Web2131 tags, our vocabulary selection method should select tags with preference toward the CAL500 tags.

[8]Note that the lack of correlation between the semantic and acoustic representation of songs does not imply that there are no "acoustically meaningful" tags. An alternative audio representation or non-linear correlation model may be used to discover a relationship with the audio signal. These variations or extensions are beyond the scope of this paper.

In the experiment, we analyzed the intersection of songs from the Web2131 and CAL500 data sets. This resulted in 363 songs with annotations from both tag data-sources. Then we performed a series of sparse CCA vocabulary selection trials to generate a sequence of vocabularies of monotonically decreasing size. Mathematically, this is done by sweeping the sparsity parameter corresponding to the semantic space, $\tilde{\rho}_{\mathbf{y}}$, from 0 (no sparsity applied) to α for some $\alpha > 0$. (Since we are not imposing sparsity on the audio feature space, we set $\tilde{\rho}_{\mathbf{x}}$ to zero.) We arbitrarily stopped at a vocabulary size of 50. For each vocabulary that is generated we record how many CAL500 tags comprise it compared to Web2131 tags.

Table I.1 shows our results and confirms our expectations. The first column in Table I.1 shows the starting vocabulary with no vocabulary selection applied. The number of tags from each vocabulary source is initially different, with the CAL500 tags comprising only 36% of the 488 tags in the combined vocabulary. Subsequent columns in the table show the state of the vocabulary as a smaller number of tags is selected. If tags from both vocabularies had an even distribution of acoustically correlated tags then we would expect the percentage of CAL500 tags to stay near a constant 36% across all columns of the table. However, the clear trend is that more tags from the CAL500 vocabulary are selected as the vocabulary size shrinks, to the point that when the vocabulary size is restricted to just 50 tags, CAL500 tags comprise 78% of the vocabulary.

If we assume that the CAL500 tags contain more acoustically descriptive tags, which we believe we have argued is likely, then our results show that the vocabulary selection method consistently chooses higher quality tags. In practice one will not know *a priori* which tags are acoustically descriptive, hence, this vocabulary selection technique can be used to discover descriptive tags in an automatic fashion.

I.5.B Vocabulary Selection for Music Retrieval

In this experiment we show how sparse CCA can be used to select acoustically meaningful tags that, as a consequence, improve the performance of a music autotagging

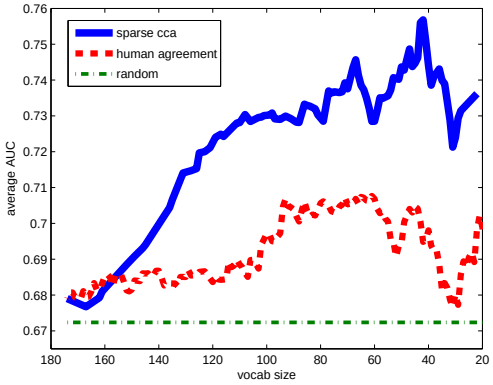

Figure I.1: Comparison of vocabulary selection techniques: We compare vocabulary selection using human agreement, acoustic correlation, and a random baseline, as it effects retrieval performance. Acoustic correlation is able to significantly increase the performance of the autotagging system.

system. Previously, we implemented a content-based music autotagging system that can annotate songs with tags from a fixed vocabulary, and retrieve songs based on a keyword query. A full description of the system can be found in a previous publication [48], although we briefly describe it here.

The system relies on learning the conditional probability distributions of audio features given tags, $P(audio|tag_i)$. This is done by extracting audio feature vectors and annotation vectors (similar to the ones used in this paper) from a training set of song-tag associations. The conditionals are learned by fitting Gaussian mixture models to the data. [9] This results in knowledge of what audio feature values are likely for a given tag. From Bayes' rule, we also have access to the conditional $P(tag_i|audio)$, in other words, knowledge of which tags are likely for given audio features. Knowing these two types of probabilities for each tag allows us to annotate a novel song with its most likely tag, and retrieve the likeliest song (from a set of novel songs) for a given keyword query (i.e., a tag). This is done by maximizing the conditionals over tags or songs depending

[9] This will be referred to as the training step or the modeling step.

on the task.

Sparse CCA can be used to select a subset of tags that is well correlated with audio content. Because of this underlying correlation pattern, we expect that focusing on these types of tags will improve the modeling power of a supervised music auto-tagging system. Therefore, we propose to use sparse CCA vocabulary selection as a pre-processing step to select high quality tags and boost the overall performance of the system.

Before we discuss the results, we must understand how to characterize the performance of the system. We test the retrieval capabilities of the system as follows: Given a test set of songs (not used to train the model) and a query tag, we rank order the songs according to their likelihood under the conditional probability $P(audio|tag_i)$. Then we evaluate the retrieval performance for this tag by calculating the area under the receiver operating characteristic curve (AUC). A receiver operating characteristic (ROC) curve is a plot of the true positive rate as a function of the false positive rate as we move down the ranked list of songs. That is, at any position in the ranking, we count the fraction of test set songs associated with the tag that are ranked at that position or higher (true positive rate) and the fraction of test set songs *not* associated with the tag that are ranked at that position or higher (false positive rate). The AUC ranges between 0.5 for a random ranking and 1.0 for a perfect ranking (i.e., all test set songs associated with the tag ranked before all that are not). The AUC gives us the retrieval performance for a *single* tag. In order to characterize the performance of the entire system, that is, over all tags, we take the average of the AUC over each tag, or *average AUC*. Note that the average AUC of the system is brought down by tags that are modeled poorly by a probabilistic acoustic model, i.e., those tags with an AUC of near 0.5. We use sparse CCA vocabulary selection to select smaller vocabularies of better acoustically-represented tags, which we expect to result in a better (larger) average AUC.

In this experiment, we use sparse CCA to generate a sequence of vocabularies that monotonically decrease in size.[10] As explained in Section I.5.A, this is done by

[10]For this experiment we use the CAL500 data set. 450 songs are used in the vocabulary selection and

sweeping the CCA sparsity parameter, $\tilde{\rho}_{\mathbf{y}}$, over a range of values. (Since we are not imposing sparsity on the audio feature space, we set $\tilde{\rho}_{\mathbf{x}}$ to zero.) For each vocabulary generated, we learn the conditional probability densities associated with each tag and then evaluate the performance of the system by calculating the average AUC over all tags in the specific vocabulary.

Figure I.1 is a graph depicting the average AUC scores of the autotagging system over a range of vocabulary sizes. The x-axis specifies the size of the vocabulary used by autotagging system. Shown in the graph are three methods for performing vocabulary selection: Vocabulary selection by acoustic correlation, by random uniform selection, and by a human agreement heuristic which we will explain below.

Looking at the acoustic correlation curve, Figure I.1 shows that the average AUC score quickly increases as the vocabulary size is reduced, confirming our expectations. This performance increase is likely due to two related factors. First, for each new vocabulary generated, a subset of tags exhibiting acoustic correlation is retained. As mentioned above, such correlation patterns are likely to allow better fit acoustic models for that set of tags. Second, tags that contribute little to the acoustic correlation are removed from subsequent vocabularies. These could be tags for which the underlying model is genuinely noisy. For example, the tag could be an error, or it could describe elements of the music that are not captured by the audio features. These noisy tags are the kinds of tags that deteriorate the average AUC system performance.

Also shown in Figure I.1 are the system's average AUC scores which have been generated using alternate vocabulary selection techniques. The flat dashed line shows the expected average AUC score if we select a vocabulary at random. The other vocabulary selection technique is an intuitive heuristic based on human agreement [44]. Because the CAL500 data set contains multiple human annotations per song, we are able to gauge how consistently our population labels music. Our intuition is that if many people use the same tag in the same manner to describe any given song, i.e., there is a high level of agreement for that tag, then this indicates some underlying audio struc-

model building steps, and 50 test songs are used to evaluate the final models.

ture that exists among songs annotated with that tag. Our acoustic models may be able to capture this structure.

To capture this idea mathematically we devised a simple statistic that we refer to as *human agreement*. First, for each tag-song pair (t, s), we calculate an individual tag-agreement score as

$$A_{t,s} = \frac{\#(\text{positive associations})_{t,s}}{\#(\text{annotations})_s}, \tag{I.20}$$

where the numerator indicates how often song s was positively associated with tag t, and the denominator indicates how often song s was presented to a human for annotation purposes. For example, if 3 out of 4 students label Elvis Presley's 'Heartbreak Hotel' as being a 'blues' song then $A_{\text{'blues', 'heartbreak hotel'}} = 0.75$. We calculate the human agreement score for a tag by averaging individual tag-agreement scores over all the songs in which at least one subject has used the tag to annotate the song. Intuitively, we expect the human agreement score to be close to 1 for more objective tags such as tags associated with instrumentation (like 'cow bell') and closer to 0 for tags that are subjective such as those relating to song usage (like 'driving music').

We believe that the human agreement statistic is a good baseline with which to compare because it is a way of selecting objective tags directly from the pool of human provided annotations. Our results show that vocabulary selection using sparse CCA is preferred over human agreement given the task of increasing the system performance.

We must emphasize that the goal of this experiment is not to improve the performance of some arbitrary system. Rather, the performance gains that we see imply that sets of tags are being selected which have a strong representation in the audio feature space. Such tags are especially suitable for machine analysis.

Our results show that vocabulary selection using sparse CCA can be used as a pre-processing step in music analysis systems whereby one can discover useful tags to model. This is especially helpful in situations where analyzing data for all available tags is a costly operation. Take, for example, our own music autotagging system. As discussed earlier, this system relies on computing conditional probability density functions

Table I.2: Top and bottom 3 tags within semantic categories according to sparse CCA vocabulary selection.

Top 3 tags by semantic category	
overall	rapping, at a party, hip-hop/rap
emotion	arousing/awakening, exciting/thrilling, sad
genre	hip-hop/rap, electronica, funk
instrument	drum machine, samples, synthesizer
general	heavy beat, very danceable, synthesized texture
usage	at a party, exercising, getting ready to go out
vocals	rapping, strong, altered with effects
Bottom 3 tags by semantic category	
overall	not weird, not arousing, not angry/aggressive
emotion	not weird, not arousing, not angry/aggressive
genre	classic rock, bebop, alternative folk
instrument	female lead vocals, drum set, acoustic guitar
general	constant energy level, changing energy level, not catchy
usage	going to sleep, cleaning the house, at work
vocals	high pitches, falsetto, emotional

for each tag in the vocabulary. Calculating these functions is generally time intensive, therefore, methods that help us avoid modeling poorly performing tags can be used to significantly speed up system efficiency.

I.5.C Qualitative Discussion

In this section, we provide some qualitative discussion related to the previous experiment. Our goal is to show that vocabulary selection using sparse CCA makes intuitive sense and passes some baseline credibility tests. Table I.2 shows the top three

tags for each semantic category where the tags have been ranked by the reverse order in which they left the vocabulary in our previous experiment. In other words, to generate these rankings we pruned the initial vocabulary of tags in a series of vocabulary selection trials. This was done by sweeping the sparsity parameter $\tilde{\rho}_{\mathbf{y}}$ from zero to some large value that yielded a null vocabulary. As mentioned above, each step generates a vocabulary of monotonically decreasing size. We took note of the order in which tags left the vocabulary and ranked them in reverse order.

Many terms that are characteristic of rap, hip-hop and electronic music are found to be at the top of the rankings. This is encouraging since both our autotagging system, as well as Mandel and Ellis' autotagging system [24], perform well on these tags. Specifically, Mandel and Ellis report that "rap","hip hop", and "techno" are 3 of the top 7 performing tags (in term of classification accuracy) in their vocabulary of 43 tags. From a more qualitative perspective, rap and electronic songs tend to have easily recognizable timbres due to their unique instrumentation (e.g., turntables, drum machines, computers) and distinct vocal characteristics (or lack thereof).

The bottom three tags per category are also shown at the bottom of Table I.2. One might suggest that these tags are ambiguous compared to many of the other tags. Consider tags such as "not weird", or "not aggressive"; we argue that these tags can be used to describe a large portion of highly varied and acoustically unrelated music, which makes these tags unlikely to contribute much to the acoustic correlation associated with a vocabulary.

I.6 Discussion

In this paper, we proposed using a novel formulation of sparse CCA to select a set of tags which is correlated with an audio feature representation. We have shown that this technique selects "acoustically meaningful" tags (Section I.5.A) and that we are better able to model these tags with our content-based autotagging system (Section I.5.B).

More generally, it should be noted that our goal is not to increase the performance of a specific autotagging system (since one could trivially increase system performance by selecting tags *a posteriori* that perform well under the performance measure). Rather, we are interested in finding a set of tags that are well represented by a given audio representation.

There are many reasons for doing this: First, selecting a subset of well-represented tags from an initial vocabulary allows us to focus computational resources on high quality tags, rather than waste time generating poorly performing tag-models. This is especially important when we consider the fact that, e.g., Last.fm has collected a database of over 1.2 million unique tags for music [19].

Second, considering that there are many audio feature representations (e.g., Fluctuation Patterns [29], Auditory Filterbank Temporal Envelopes [26], Beat Histograms [50], Beat-Synchornous Chroma vectors [8]), we may be interested in finding at least one representation for which our vocabulary selection method chooses some tag for it. If this tag is never selected, we may need to focus on context-based approaches (e.g., surveys, video games, social tagging) to model that tag [46].

Third, we are often interested in designing user interfaces for music search and discovery (e.g., Last.fm, MusicSun [28]) that make use of tags for visualization and navigation purposes. By finding a small set of acoustically meaningful tags, we can produce a concise representation of music that enriches the user experience.

While our sparse CCA algorithm can be used to find some high quality tags, it is unlikely that it finds them all. For example, relevant non-linear acoustic patterns may exist which the current sparse CCA algorithm would not capture. A "kernelized" version of the CCA algorithm [33] is a promising method for exploring non-linear cross-correlations. However, imposing sparsity in kernelized algorithms often implies that we must sacrifice the interpretability of variable selection. Fortunately, since our main interest is to perform variable selection in the semantic space (i.e., select a subset of tags), losing interpretability in the audio feature space can be tolerated. This would allow us to kernelize the audio feature space in an attempt to discover non-linear correlations be-

tween feature spaces. This avenue for future work is non-trivial because, due to the large number of audio feature vectors generated by our music data, the corresponding kernel matrices become extremely large; hence, highly efficient, large-scale kernel-based approaches should be investigated.

This chapter, in full, has been submitted for publication of the material as it may appear in IEEE Transactions on Audio, Speech and Language Processing 2009. Torres, David; Sriperumbudur, Bharath; Turnbull, Douglas; Lanckriet, Gert R. G., 2009. The thesis author was the primary investigator and author of this paper.

References

[1] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004.

[2] P. S. Bradley and O. L. Mangasarian. Feature selection via concave minimization and support vector machines. In *Proc. 15th International Conf. on Machine Learning*, pages 82–90. Morgan Kaufmann, San Francisco, CA, 1998.

[3] A. d'Aspremont, L. El Ghaoui, M.I. Jordan, and G.R.G. Lanckriet. A direct formulation for sparse pca using semidefinite programming. *SIAM Review*, 49:434–448, 2006.

[4] Tijl De Bie, Nello Cristianini, and Roman Rosipal. Eigenproblems in pattern recognition. *Handbook of Computational Geometry for Pattern Recognition*, 2004.

[5] J. S. Downie. Audio tag classification task. Music Information Retrieval Evaluation eXchange (MIREX), 2008.

[6] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2000.

[7] D. Eck, P. Lamere, T. Bertin-Mahieux, and S. Green. Automatic generation of social tags for music recommendation. In *NIPS*, 2007.

[8] D.P.W. Ellis and G.E. Poliner. Identifying cover songs with chroma features and dynamic programming beat tracking identifying 'cover songs' with chroma features and dynamic programming beat tracking. *IEEE ICASSP*, 2007.

[9] S. Essid, G. Richard, and B. David. Inferring efficient hierarchical taxonomies for music information retrieval tasks: Application to music instruments. *ISMIR*, 2005.

[10] Y. Freund and R. Schapire. A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence*, 1999.

[11] D. R. Hardoon and J. Shawe-Taylor. The double-barrelled lasso. *Learning from Multiple Sources Workshop, NIPS*, 2004.

[12] D. R. Hardoon, J. Shawe-Taylor, and O. Friman. Canonical correlation analysis: An overview with applications to learning methods. *Technical Report, University of London*, 2003.

[13] D.R. Hardoon, C. Saunders, S. Szedmak, and J. Shawe-Taylor. A correlation approach for automatic image annotation. *The 2nd International Conference on Advanced Data Mining and Applications*, 2006.

[14] T. Hastie, R. Tibshirani, M. B. Eisen, A. Alizadeh, R. Levy, L. Staudt, W. C Chan, D. Botstein, and P. Brown. 'gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology*, 2000.

[15] R. Horst and N. V. Thoai. D.c. programming: Overview. *Journal of Optimization Theory and Applications*, 1999.

[16] X. Hu, J. S. Downie, and A. F. Ehmann. Exploiting recommended usage metadata: Exploratory analyses. 2006.

[17] I.T. Jolliffe, N.T. Trendafilov, and M. Uddin. A modified principal component technique based on the lasso. *Journal of computational and graphical statistics*, 2003.

[18] Einat Kidron, Yoav Y. Schechner, and Michael Elad. Pixels that sound. In *IEEE Computer Vision and Pattern Recognition*, 2005.

[19] P. Lamere and E. Pampalk. Social tags and music information retrieval. ISMIR Tutorial, 2008.

[20] M. Levy and M. Sandler. A semantic space for music derived from social tags. In *ISMIR*, 2007.

[21] Tao Li and George Tzanetakis. Factors in automatic musical genre classification of audio signals. *IEEE WASPAA*, 2003.

[22] Yaoyong Li and John Shawe-taylor. Using kcca for japanese-english cross-language information retrieval and classification. *Journal of Intelligent Information Systems*, 27:117–133, 2005.

[23] Beth Logan. Mel frequency cepstral coefficients for music modeling. *ISMIR*, 2000.

[24] M. Mandel and D. Ellis. Multiple-instance learning for music information retrieval. In *ISMIR*, 2008.

[25] C. McKay, D. McEnnic, and I. Fujinaga. A large publically accessible prototype audio database for music research. *ISMIR*, 2006.

[26] M.F. McKinney and J. Breebaart. Features for audio and music classification. In *ISMIR*, 2003.

[27] F. Pachet and P. Roy. Hit song science is not yet a science. *ISMIR*, 2008.

[28] E. Pampalk and M. Goto. Musicsun: A new approach to artist recommendation. *ISMIR*, 2007.

[29] Elias Pampalk. *Computational Models of Music Similarity and their Application to Music Information Retrieval*. PhD thesis, Vienna University of Technology, 2006.

[30] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 1999.

[31] L. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, 1993.

[32] J. Reed and C.H. Lee. A study on attribute-based taxonomy for music information retrieval. *ISMIR*, 2007.

[33] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[34] J. Skowronek, M. McKinney, and S. ven de Par. Ground-truth for automatic music mood classification. 2006.

[35] Malcolm Slaney. Semantic-audio retrieval. *IEEE ICASSP*, 2002.

[36] M. Sordo, C. Lauier, and O. Celma. Annotating music collections: How content-based similarity helps to propagate labels. In *ISMIR*, 2007.

[37] B. K. Sriperumbudur, D. A. Torres, and G. R. G. Lanckriet. The sparse eigenvalue problem. *http://arxiv.org/abs/0901.1504v1*, January 2009.

[38] Bharath K. Sriperumbudur, David A. Torres, and Gert R. G. Lanckriet. Sparse eigen methods by d.c. programming. In *International Conference on Machine Learning*, 2007.

[39] Bharath K. Sriperumbudur, David A. Torres, and Gert R. G. Lanckriet. The sparse eigenvalue problem. *http://arxiv.org/abs/0901.1504*, 2009.

[40] P. D. Tao and L. T. H. An. D.c. optimization algorithms for solving the trust region problem. *SIAM Journal on Optimization*, 1998.

[41] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 1996.

[42] M. Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 2001.

[43] D. Torres, B. Sriperumbudur, and G. Lanckriet. Finding musically meaningful-words by sparse cca. In *NIPS Workshop on Music, the Brain and Cognition*, 2007.

[44] D. Torres, D. Turnbull, L. Barrington, and G. Lanckriet. Identifying words that are musically meaningful. In *ISMIR 07*, 2007.

[45] D. Turnbull, L. Barrington, and G. Lanckriet. Modelling music and words using a multi-class naïve bayes approach. In *ISMIR*, 2006.

[46] D. Turnbull, L. Barrington, and G. Lanckriet. Five approaches to collecting tags for music. In *ISMIR*, 2008.

[47] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Towards musical query-by- semantic- description using the CAL500 data set. In *ACM SIGIR*, 2007.

[48] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet. Semantic annotation and retrieval of music and sound effects. *IEEE TASLP*, 16(2):467–476, February 2008.

[49] D. Turnbull, R. Liu, L. Barrington, D. Torres, and G Lanckriet. Using games to collect semantic information about music. In *ISMIR*, 2007.

[50] G. Tzanetakis and P. R. Cook. Musical genre classification of audio signals. *IEEE Transaction on Speech and Audio Processing*, 10(5):293–302, 7 2002.

[51] Vladmir Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

[52] Alexei Vinokourov, John Shawe-Taylor, and Nello Cristianini. Inferring a sementic representation of text via cross-language correlation analysis. *Advances in Neural Information Processing Systems*, 2003.

[53] L. von Ahn, R. Liu, and M. Blum. Peekaboom: A game for locatingobjects in images. *ACM CHI*, 2006.

[54] J. Weston, A. Elisseeff, B. Schölkopf, and M. Tipping. Use of the zero-norm with linear models and kernel methods. *Journal of machine learning research*, 2003.

[55] B. Whitman and D. Ellis. Automatic record reviews. In *ISMIR*, 2004.

[56] D.P. Wipf and B.D. Rao. Sparse bayesian learning for basis selection. *IEEE Trans. on Signal Processing*, 2004.

[57] Y. Yamanishi, J.-P. Vert, A. Nakaya, and M. Kanehisa. Extraction of correlated gene clusters from multiple genomic data by generalized kernel canonical correlation analysis. *Bioinformatics*, 19, 2003.

[58] A. L. Yuille and A. Rangarajan. The concave-convex procedure. *Neural Computation*, 15:915–936, 2003.

[59] Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 2004.