Affective Modeling of Music using Probabilistic Features Representations

Jens Madsen, *Student Member, IEEE,* Bjørn Sand Jensen , *Member, IEEE,* and Jan Larsen, *Member, IEEE*

***Abstract*—The temporal structure in music is an essential aspect when we as humans categorize and describe the cultural, perceptual and cognitive aspects of music such as genre, emotions, preference and similarity. Historically, however, temporal information has largely been disregarded when building automatic annotation and labeling systems of music. Both in music navigation and recommendation systems. This paper addresses this apparent discrepancy between *common sense* and the majority of modeling efforts by first providing an analysis and survey of existing work, proposing a simple taxonomy of the many possible feature representations. Next, the different paths in the taxonomy are evaluated by testing the hypothesis whether it is beneficial to include temporal information for predicting high-order aspects of music. We specifically look into the emotions expressed in music as a prototypical high-order aspect of audio.**

We test the hypothesis and difference between representations using the following pipeline: 1) Extract features for each track obtaining a multivariate *feature time-series*. 2) Model each track-level time-series by a probabilistic model: Gaussian Mixture models, Autoregressive models, Linear Dynamical Systems, Multinomial models, Markov and Hidden Markov models. 3) Apply the Probability Product Kernel to define a common correlation/similarity function between tracks. 4) Model the observations using a simple, well-known (kernel) logistic classification approach specifically extended for two-alternative-forced choice to ensure robustness. The evaluation is performed on two data sets, including two different aspects of emotions expressed in music.

The result provides evidence that increased predictive performance is obtained using temporal information, thus supporting the overall hypothesis.

I. Introduction

With the ever-growing collections and online availability of music, easy and intuitive methods of accessing these large collections has become more pertinent than ever.

This has been addressed in various ways, ranging from context and collaborative approaches to purely content-based models. The focus of this work is on the content-based approach, from the audio signal itself, where the aim is to predict aspects which are of relevance in navigating and exploring music archives, such as high-order cognitive aspects like genre, emotion and perceived similarity.

Such content-based, predictive models have largely relied on three major elements: First, self-reported annotations (rankings, ratings, comparisons, tags, etc.) for quantifying the specific higher-level cognitive aspect. Secondly, finding a suitable audio representation (using audio or lyrical features), and finally associating the two aspects using machine-learning methods with the aim to create predictive models of the labels/annotations.

This traditional approach has seemingly reached a glass ceiling for predicting e.g. the emotions expressed in music, as mentioned in [1], genre prediction [2] and melody [3]. An example of the glassceiling is the MIREX Automatic Mood Classification (AMC) competition. Despite the many attempts over the years to use an increasing number of audio features and greater modeling complexity, there seems to be a 66% limit on the classification accuracy. Even later work [4] using the same taxonomy for acquiring labels, reached similar limitations. In [1] they argued that the annotation procedure, model of representing emotions and the taxonomy used are all likely to be the limiting factors, and they set a natural limit for how well a system can perform. In a similar fashion, music genre recognition also suffers from acquiring reliable genre labels [5] and sets natural limits to how well models can perform on solving the specific task.

It has been argued [3] that one reason for the performance limit is the so-called semantic gap, i.e. a fundamental gap between handcrafted, low-level audio features designed to capture different musical aspects and the actual higher-order, cognitive aspects which are of relevance for a particular task. A source for this gap can easily be identified in the way predictive systems represent the audio itself, since audio streams are often represented with frame-based features. The signal is thus divided into frames with various lengths depending on the musical aspect which is to be analyzed. Features are extraction of the enframed signal resulting in a multivariate time series of features. In order to use these features in a modern discriminative setting, they are often represented using simple pooling functions such as the mean, or a single/mixture Gaussian. This reduces the full time series to a single vector which is applicable in traditional linear models or kernel machines, e.g. Support Vector Machines (SVM). The main problem is that this approach disregards all temporal information in the extracted features.

Some work has gone into examining temporal integration [6], and this has shown an improved performance on genre prediction and emotion recognition [7]. In [7] we proposed specifically extending the audio representation by including a *feature representation* as an additional aspect to consider, as illustrated in Figure II-G2. We proposed a common framework to code both temporal and non-temporal aspects in discrete and continuous features using generative models. This both unified and extended previous work in how to code features for creating predictive models. We see that an important step to narrowing the semantic gap and potentially breaking through

This work was supported in part by the Danish Council for Strategic Research of the Danish Agency for Science Technology and Innovation under the CoSound project, case number 11-115328.

the glassceiling is to add a layer of temporal modeling to the low-level handcrafted features.

Our aim in this work is to outline and summarize state-of-the art in representing the audio in music with and without temporal integration, and subsequently evaluate the potential benefits in including temporal information for current and relevant prediction tasks. In order to ensure a fair comparison of the representations, we suggest a common model for comparing the representation based on probabilistic representations and a particular kernel-based model.

In the evaluation, we consider the emotions expressed by music, a prototypical example of a higher-level cognitive aspect. Music's ability to represent and evoke emotions is an attractive and yet a very complex quality for navigating and searching music archives.

This extends the work from [7] by exploring the use of a number of additional generative models as the feature representation, and using these models on five, often-used, handcrafted, low-level features showing a significant performance gain from using these models.

The organization of this paper is as follows. In Section II a broad background of the content-based modeling pipeline is presented. In Section III the proposed framework for feature representation is presented. In Section IV the pairwise kernel logistic regression model used for incorporating the feature representations is presented. In Section **??** the dataset and the evaluation methods are presented. In Section V we present the results of the proposed feature representations, evaluated on pairwise emotion comparisons. In Section VI we discuss the results and lastly in Section VII we conclude on our findings.

II. Background

In this section, we give a broad overview of the elements involved in creating predictive models of higher-level aspects of music such as genre, tags and expressed emotions based on the audio signal itself. We use the term *audio representation* as a placeholder for all the approaches used to represent the audio in a mathematical form when going from the digitized audio signal to finally serve as an input to a predictive model (i.e. regression or classification).

The following overview is based on the informal taxonomy given on Fig. II-A. At the lowest level, the audio is represented in either the temporal or spectral domain, as illustrated on Fig. II-A. These domains are naturally used interchangeably throughout MIR and are used as the basic input for the two major directions. The first approach (*feature extraction*) extracts low-level/handcrafted features, designed to capture different aspects of the audio signal using both the temporal and spectral domain (Section II-B). The second approach uses either the time-signal or the tempo-spectral representation of the audio directly, typically using e.g. the spectrogram (Section II-C). Common for both approaches is that many (either implicitly or explicitly) find compact representations of the continuous data using either subspace methods (Section II-D) or discretize the basic representations, such as the spectrogram, to find meaningful (or at least computationally tractable) representations (Section II-E).

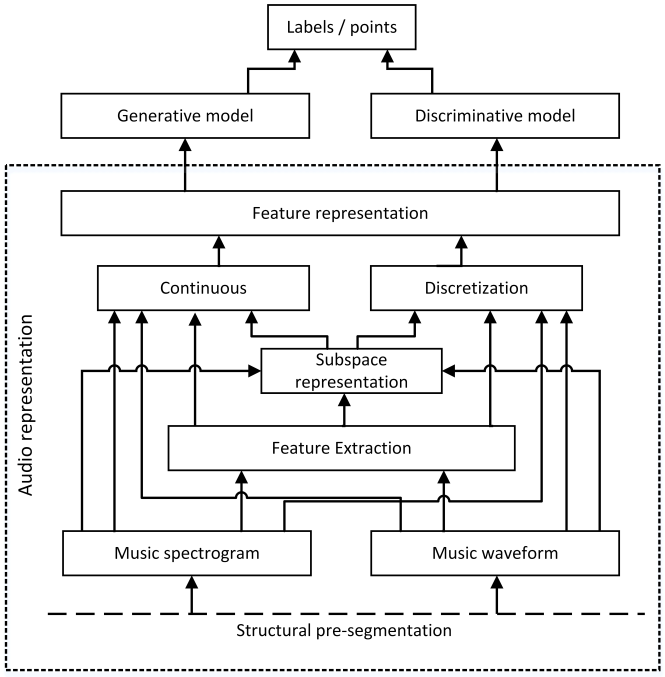

Figure 1. System overview.

Regardless of the approach, the result is a time series of either discrete or continuous values, which is summarized using an appropriate feature representation (Section III). Finally, the feature representation serves as an input to the predictive model (Section II-G), e.g a SVM or the final layer in a DNN.

We note that some methods, such as Deep Neural Networks integrate many of the elements into one structure, however, for the sake of presentation, we differentiate between the representation (e.g. lower layers) and the prediction (last layer) in order to provide a unified overview.

A. Pre-segmentation

Pre-segmentation is often used prior to any feature extraction or representation to account for the naturally segmented structure of music. The segmentation is also used to capture some of the temporal evolution in music e.g. [8], [9], [10], [11], [12] using fixed-sized windows as [13] introduced as *texture windows*. The size of the texture windows can be optimized in this pre step as in [14] using e.g. boosting. Late integration techniques can be used on top of this pre-segmentation to obtain a single label as an output of a predictive model of e.g. genre, tags or emotions (see Section II-F1 for more about integration methods).

B. Feature extraction

A great deal of work has been put in to automatically extract features which are often found in the musical literature and used by musicologists in their work, such as tempo, onsets, beat and downbeats, multiple fundamental frequencies, key and chord extraction (MIREX tasks[1]). These hand-crafted

[1] http://www.music-ir.org/mirex/wiki

features have been the building blocks of the machine-learning methods used to predict the higher-order aspects of music for a long time. The often-used approach for low-level features is the frame-based digital signal processing approach, whereby the waveform is windowed into overlapping frames and different musical aspects are extracted. Some attempts have been made to build temporal information across frames into the features themselves, e.g. using fluctuation patterns [15] or in the multi-timescale Principal Mel-Spectrum Components by using simple summary statistics [16].

For the specific task of creating predictive models of emotions in music, the approaches have been agnostic in feature selections and often gathered a great deal of different features and let the predictive performance guide which features were appropriate e.g. [17]. Multiple combinations have since been used and no greater insight seems to have surfaced [18], [1] into which features to use.

C. Spectrogram

The spectrogram is often used as a basic representation of the audio signal, typically followed by an unsupervised machine-learning technique, for example to find subspaces (see II-D) and/or sparse coding to find key components [19]. An often-realized option is to compute tempo-spectral representations which align with the human perceptual and cognitive understanding. Various transforms are used to this end, such as log scaling of the frequency bands e.g. mel-scaled spectrogram [9], [20] or using a constant-Q transform [10], [8].

D. Subspace representation

A subspace representation effectively reduces the dimensionality of the original signal/feature space and provides a more compact representation of the signal/features, either with the purpose to provide an interpretable and meaningful view on low-level features or spectrogram - or simply functioning as a more practical and compact representation, potentially increasing the performance of subsequent modeling steps.

1) Spectrogram: The raw spectrogram potentially captures redundant and irrelevant information for the task at hand. To filter this irrelevant information and empathize/extract the important aspect of the signal, many approaches are used to find the underlying informative subspace which potentially increases the performance of the subsequent predictive model, such as principle component analysis, non-negative matrix factorization or independent component analysis. These methods are also used as a step towards discretization using dimensionality-reduction techniques like PCA [21]. Furthermore, sparse versions/extensions have been widely used for e.g. music annotation [22], tag prediction [20], genre prediction [19] and sound-effect retrieval [23].

With the evolution of artificial neural networks and the ability to use highly parallel computing, neural networks have grown with multiple layers into so-called deep architectures, often trained in a mixture between unsupervised and supervised approaches. The first layers of neural network, e.g. trained with unsupervised, restricted Boltzmann machines, autoencoders, can be seen as a projection of the input features onto a subspace (like on Fig. II-A), defined by vectors represented by the weights and activation functions on the individual nodes in each layer. Hence, the neural network finds a non-linear/linear subspace given the specific architecture in a similar fashion to the other subspace methods, albeit sometimes in a supervised fashion. We here separate the decision part and the purely unsupervised part of the network, although these are often an integrated architecture.

Various combinations of supervised and unsupervised neural networks have been used in many fields, starting with speaker, phone and gender recognition e.g. [24], and there have been adopted for MIR tasks, e.g. for genre prediction [25], instrument detection [26], artist recognition [27], music similarity [28] and for finding structures useful in emotion prediction [29].

The temporal aspect can be incorporated implicitly into neural networks using e.g. convolutive and recurrent neural networks. This has been done for speech recognition in [30] using recurrent neural networks, however within MIR it has not yet been adopted widely. In MIR, the typical approach is still to analyze and predict on texture windows of a certain length and do late decision/majority voting effectively in a different step.

2) Low-level features: The dimensionality of the extracted low-level features is often high, and PCA, NMF, PKLS [31], ICA and other dimensionality-reduction techniques are used as a step prior to the feature representation [28] or explicitly to decorrelate features [32][24], [20]. DNNs are also used in this context to find suitable subspace representations of low-level features, as used by [27] for genre prediction using Echonest[2] features provided by the Million Song Dataset[3].

E. Discretization

The output of the feature extraction (or the signal itself) can be both discrete or continuous, as shown on Fig. II-A. A particularly computationally efficient way of representing the audio is through discretization in which the continuous time-series (e.g. spectrogram or low-level features) are encoded as belonging to a finite set of codewords in a given codebook. This discretization is divided into two steps, namely defining the codewords and subsequently assigning features/signal to a finite number of these codewords:

I Codebook construction: Codebook construction is traditionally done in a multitude of different ways, depending on the task and input (see II-A):

- **Manually** defining or fixing the basis functions is possible, e.g. for the raw audio signal or low-level features, and typically entails a sinusoidal, gammatone [33], wavelets or Gabor basis resulting in a traditional spectral transform [4]

[2]http://developer.echonest.com/acoustic-attributes.html
[3]http://labrosa.ee.columbia.edu/millionsong/
[4]We note that the outline in figure TODO supports

- **Learning** the cookbook from data - either the current corpus or an independent one - is typically done using the aforementioned subspace methods such as PCA, NMF, k-means, sparse coding, or LASSO.
- **Exemplar** -based cookbook is constructed from the observed data, and has shown to give similar performance to learning the dictionary [10].
- **Random** projections in which the basis is simply randomly initialized have also been used for music application, e.g. [34], although not as widely as e.g. NMF or similar.

II Encoding: The actual process of discretization, i.e. assigning the signal/features to one or more code words in a binary manner, varies from simple threshold/max to solving sparse/dense coding problems [9]. Top-τ vector quantization was introduced in [35], assigning the τ-nearest codewords to each frame. Interestingly enough, increasing τ to a certain level makes the discrete encoding more robust and results in better performance in query-by-tag and query-by-example tasks. In this setting, standard vector quantization is obtained for $\tau = 1$ and k-means with euclidean is what is traditionally referred to as vector quantization (VQ)

1) Low-level features: Codebooks are computed using standard methods e.g. k-means [35], [36], [37], [20] and LASSO [23], [35] on MFCC features, with different distance measures (euclidean, cosine similarity etc.). Sparse version has been applied for tag prediction and genre recognition on MFCC [10], [20] and Sonogram [10] and Principle Mel-spectrum components [9] for genre recognition. Another approach is to see the parametrization of different generative models trained on music excerpts or segments as a word and using the likelihood as a distance function between an excerpt and the generative models. The codebook is now the parametrization of generative models, if the feature representation is a simple frequency-based representation, i.e. counting the frequency of each model, then this results in what the authors call a Bag-of-Systems (BoS) [38]. This approach can work on different timescales e.g. sizes of texture windows, and therefore the number of words for each song can vary.

2) Spectrogram: A more efficient representation of the spectrogram can be found via a dense and/or sparse basis via e.g. k-means or sparse coding. K-means is a fast method of finding cluster centers, e.g. in the context of codebook construction-dense codewords. In part due to the sheer speed of computation, this method has been used extensively e.g. for tag prediction [20], genre classification [39], and it has been expanded to use patches of the spectrogram in [40]. One popular method is using Sparse Coding (SC) for genre classification [10], [21], [9], [8], instrument recognition and tag prediction [9], [20]. Common for these approaches is that they largely disregard the temporal aspects of the codewords.

F. Feature representation

Common, for most of the approaches described above, is an output which still has a temporal dimension, i.e. a time series of features, codewords, subspace projection (from e.g. a neural network at a particular layer), or spectral representation.

Regardless of whether the output of previous steps is continuous or discrete, the new time series has to be summarized over time, allowing one to map the potentially variably sized vector into a representation that can be fed to a classifier. Some methods often used to summarize this representation are *temporal pooling*, here used to encompass a vast amount of methods.

1) Temporal pooling/integration: We define a temporal pooling function as any function that is able to transform the time series of the features (or annotation) into a more temporally compact representation. This can be done using e.g. summary statistics or probabilistic models encoding the temporal structure of the time-series. When frame-based analysis is used, a texture window is represented by the chosen pooling function. If the annotation is on the entire track and not on each texture window, a method of integrating the decision to achieve one single prediction is required e.g. tag, genre or emotion. This process is often referred to as **late integration** [41], where methods like decision fusion (e.g. majority voting [6]), kernels (e.g. convolutive or alignment kernels [41][42]) or HMM can be used [41]. Depending on whether the features are discrete or continuous, different types of strategies are used for early integration/temporal pooling, summarizing the features locally.

Discrete: A discrete time series (e.g. following discretization) of features or a spectrogram has to be somehow represented before being input to a model. A very popular way is to obtain the frequency of each discrete entry and represent this as a histogram, which in most literature is called *Bag-of-Features* or *Bag-of-Frames* representation. Due to the simple counting, the representation neglects all temporal information in the time series. Methods have been attempted to account for some temporal content using e.g. the texture windows [9] and using so-called *Bag-of-Histograms* where a BoF representation is obtained for each texture window, however temporal information is not coded locally within each window. The term 'pooling functions', first used in image processing, has also been adapted in the audio community e.g. [43][21]. Here a multitude of functions have been proposed for summarizing features across time e.g. average, max, log, energy, magnitude, cuberoot, etc. In [44] they use string compressibility as a summary statistic for song year prediction and music similarity showing improved performance compared to traditional summary statistics. In previous work [7] we proposed using generative models to code temporal content in discrete data for emotion prediction. Here Markov and Hidden Markov models were trained on each track and used in a discriminative setting using kernel methods.

Continuous: The combination of using summary statistics and texture windows is a popular way of summarizing some temporal information using e.g. mean [12] and standard deviation [13] forming a single-vector representation or exploring other simple statistics and window sizes [11][14]. Variants of this also propose using temporal feature stacking using a lag window to further account for temporal evolution through music [45][29]. The use of generative models to obtain a representation on the track level using non-temporal models such as the GMM has been proposed by [46][47][48], treating the

features as Bag-of-Frames. Some of the first work accounting for temporal structure using generative models was [42][6] where the AR model was used to summarize each texture window for genre prediction. This idea was continued in [41] for instrument classification exploring different methods of early and late integration. One drawback of using kernel-based methods is the scaling, especially when a great number of texture windows are used, since the kernel evaluations grow quadratically. This issue was addressed specifically when using the AR model in [49], where instead of a distance between all AR models fitted, a mixture of AR models was proposed to reduce the number of kernel evaluations.

G. Modeling

In this section we review some of the methods used to take temporal aspects of the musical signal into account on the modeling side of creating predictive models of higher-order cognitive categorization of music as e.g. tags, genre and emotions.

1) Generative: A very popular generative model taking the sequence of features into account is the HMM, often used in speech recognition, which has been adapted for use on classic MIR tasks. Segmentation of audio [50] and chords [51] were amongst the first to be adopted. Later came genre recognition [52][53] and key estimation [54]. Extending to a non-parametric treatment of the classic HMM, [55] used the Hierarchical Dirichlet process to select the number of states in the HMM, modeling the temporal evolution of music. Linear dynamical systems or Dynamic textures has also been used for segmentation [56], where they extend the classic model to include a mixture of DTs (DTM). In [57] they showed that using the DTM model to represent the feature time series of MFCCs, taking temporal dynamics into account, carried a substantial amount of information about the emotional content.

2) Discriminative: The discriminative models are by far the majority of models used in MIR, where SVM/SVR, K-NN and traditional linear methods like RLS are very commonly used [58][59]. Some of the methods used to include temporal information about the audio using discriminative models include using the probability product kernel between AR models fitted to excerpts [49][42][7] thus using the generative models in a discriminative setting. In modeling the emotions expressed in music, the temporal aspect of emotion has been centered on how the labels are acquired and treated, not on how the musical content is treated. E.g. in [60] they used a Conditional Random Field (CRF) model to essentially smooth the predicted labels of an SVM, thus still not providing temporal information regarding the features. In [12] a step to include some temporal information regarding the audio features was made by including some first and second order Markov properties for their CRF model, however still averaging the features for one-second windows.

H. Present work

In the present work, we focus on creating a common framework for evaluating the importance of temporal information using generative models as feature representation for multivariate-feature time series. In particular, we focus the evaluation on modeling aspects related to the emotions expressed in music. Since very little work has been done on evaluating temporal integration within this field, we make a broad comparison of a multitude of generative models of time-series data.

We distinguish between how the time series are modeled on two aspects: whether the time series are continuous or discrete, and whether temporal information should be taken into account or not. This results in four different combinations, which we investigate:

1) **Continuous**, temporally **independent** representation: using mean, single Gaussian and GMM models.
2) **Continuous**, temporally **dependent** representation: using Autoregressive models, Linear Dynamical Systems (LDS) and Hidden Markov Models with Gaussian emissions (HMM_{cont}).
3) **Discretized**, temporally **independent** representation: using vector quantization in a Bag-of-Audiowords model.
4) **Discretized**, temporally **dependent** representation: using Markov and Hidden Markov Models (HMM).

A multitude of these models have never (to our knowledge) been used in MIR as a track-based representation and compared systematically. To use these generative models in a discriminative setting, the Product Probability Kernel (PPK) is selected as a natural kernel for all considered feature representations. We extend a kernel-generalized linear model (kGLM) specifically for pairwise observations for use in predicting the emotions expressed in music.

In total, nine different feature-representation models are applied on five different popular low-level features. We evaluate the features and the feature-representation models using predictive performance on two datasets of pairwise comparisons evaluated on the valence and arousal dimensions.

III. Feature Representation

In order to model higher-order cognitive aspects of music, we first consider standard audio-feature extraction which results in a frame-based, vector-space representation of the music track. Given T frames, we obtain a collection of T vectors with each vector at time t denoted by $\mathbf{x}_t \in \mathbb{R}^D$, where D is the dimension of the feature space.The main concern here is how to obtain a track-level representation of the sequence of feature vectors for use in subsequent modelling steps. In the following, we will outline a number of different possibilities — and all these can be considered as probabilistic densities over either a single feature vector or a sequence of such (see also Table. I).

Continuous: When considering the original feature space, i.e. the sequence of multivariate random variables, a vast number of representations have been proposed, depending on whether the temporal aspects are ignored (i.e. considering each frame independently of all others) or modeling the temporal dynamics by temporal models.

In the time-independent case, we consider the feature as a bag-of-frames representation, and compute moments of the independent samples; namely the mean. Including higher order

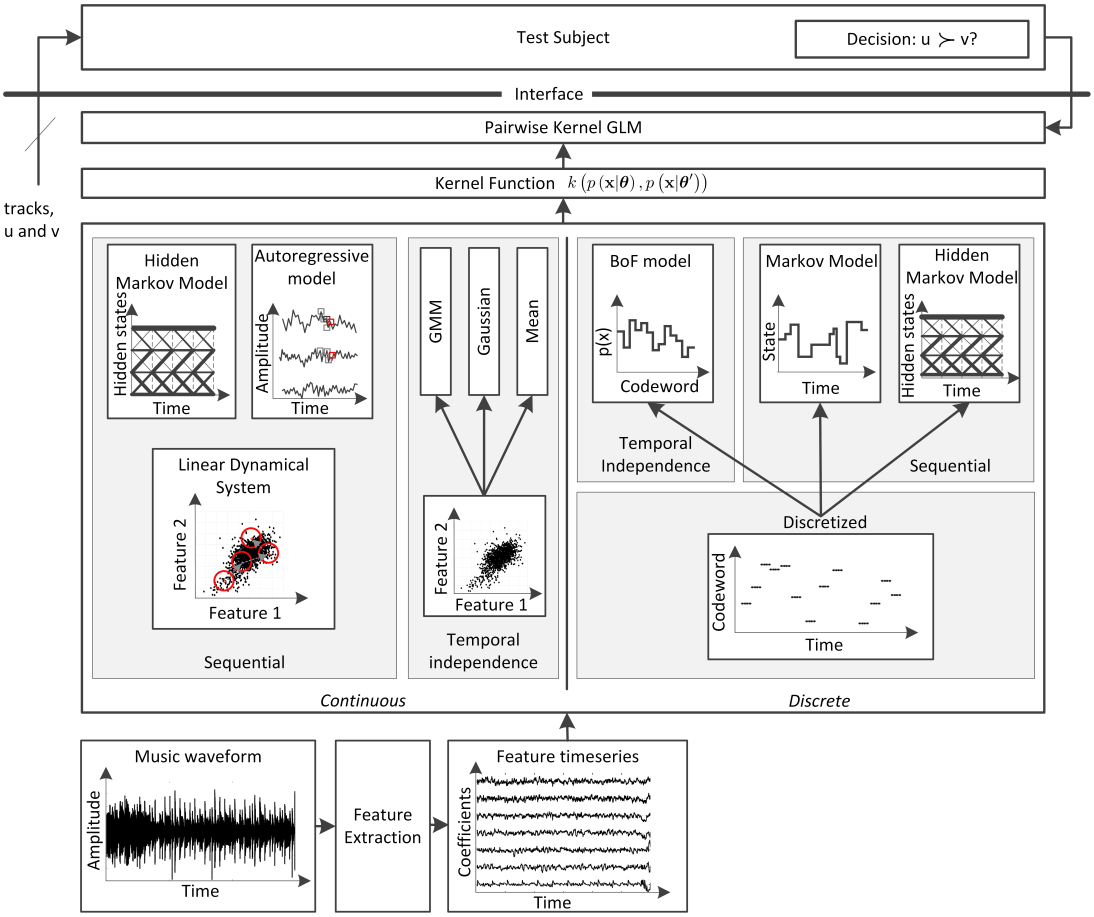

Figure 2. System overview. Starting from the bottom (left): each excerpt is represented by its temporal waveform, from which standard audio feature extraction transforms the data into a window-based multivariate vector representation. This vector representation acts as input for a multitude of statistical density models. The first decision is whether to operate in the original continuous vector space (*Continuous*) or encode the vector space using vector quantization (*Discrete*) (which implies initially operating in the continuous domain to identify codewords and perform encoding). Following this choice, the main decision is whether to encode the temporal (Sequential) aspect or not "Temporal independence". The computation of a specific density representation for all excerpts is then fed to the kernel function in order to effectively define similarity between excerpts. This is finally used in the pairwise kernel GLM model. The pairwise kernel GLM utilizes these representations to model the pairwise judgment by each subject between two excerpts in terms of their expressed Arousal or Valance.

moments will naturally lead to the popular choice of representing the time-collapsed time series by a multivariate Gaussian distribution (or other continuous distributions). Generalizing this leads to mixtures of distributions such as the GMM (or another universal mixture of other distributions) used in an abundance of papers on music modeling and similarity (e.g. [62], [63]).

Instead of ignoring the temporal aspects, we can model the sequence of multivariate feature frames using well-known temporal models. The simplest models include AR models [6]. Further extending this principle leads to Linear Dynamical Systems (LDS) [61] or with discrete states the Hidden Markov Model with e.g. Gaussian observation ($\mathrm{HMM}_{\mathrm{cont}}$). Mixtures of any of the mentioned representations may also be considered, as in [49].

Discrete: In the discrete case, features are naturally discrete or the original continuous feature space can be discretized using e.g. VQ with a finite set of codewords resulting in a dictionary(found e.g. using K-means). Given this dictionary, each feature frame is subsequently assigned a specific codeword in a 1-of-P encoding such that a frame at time t is defined as vector $\tilde{\mathbf{x}}_t$ with one non-zero element.

At the track level and time-independent case, each frame is encoded as a Multinomial distribution with a single draw, $\tilde{\mathbf{x}} \sim \mathrm{Multinomial}(\boldsymbol{\lambda}, 1)$, where $\boldsymbol{\lambda}$ denotes the probability of occurrence for each codeword and is computed on the basis of the histogram of codewords for the entire track. In the time-dependent case, the sequence of codewords, $\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_T$, can be modeled by a relatively simple (first order) Markov model, and by introducing hidden states this may be extended to the (homogeneous) Hidden Markov model with Multinomial observations ($\mathrm{HMM}_{\mathrm{disc}}$).

A. Estimating the Representation

The probabilistic representations are all defined in terms of parametric densities which in all cases are estimated using standard maximum likelihood estimation (see e.g. [61]). Model selection, i.e. the number of mixture components in the GMM, order of the AR model, number of hidden states in the HMM models and dimensionality of latent dimension in LDS, is explored using two different approaches. A global representation where model selection is performed by e.g.

Obs.	Time	Representation	Density Model	$\boldsymbol{\theta}$	Base
Continuous	Indp.	Mean	$p(\mathbf{x}\|\boldsymbol{\theta}) \equiv \delta(\boldsymbol{\mu})$	μ, σ	Gaussian
		Gaussian	$p(\mathbf{x}\|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{x}\|\mu, \boldsymbol{\Sigma})$	$\mu, \boldsymbol{\Sigma}$	Gaussian
		GMM	$p(\mathbf{x}\|\boldsymbol{\theta}) = \sum_{i=1}^{P} \lambda^{(i)} \mathcal{N}\left(\mathbf{x}\|\boldsymbol{\mu}^{(i)}, \boldsymbol{\Sigma}^{(i)}\right)$	$\{\lambda^{(i)}, \mu^{(i)}, \boldsymbol{\Sigma}^{(i)}\}^{i=1:L}$	Gaussian
	Seq.	AR	$p(\mathbf{x}_0, \mathbf{x}_1, .., \mathbf{x}_P\|\boldsymbol{\theta}) = \mathcal{N}\left([\mathbf{x}_0, \mathbf{x}_1, .., \mathbf{x}_P]^\top \|\mathbf{m}, \boldsymbol{\Sigma}_{\|\mathbf{A},\mathbf{C}}\right)$	$\mathbf{m}, \boldsymbol{\Sigma}_{\|\mathbf{A},\mathbf{C}}$	Gaussian
		LDS	$p(\mathbf{x}_0, \mathbf{x}_1, .., \mathbf{x}_T\|\boldsymbol{\theta}) = \int_{\mathbf{z}_{0:T}} \eta_{\mathbf{z}_0} \prod_{t=1}^{T} \boldsymbol{\Omega}_{\mathbf{z}_t,\mathbf{z}_{t-1}} \boldsymbol{\Upsilon}_t$	$\eta, \boldsymbol{\Omega}, \boldsymbol{\Upsilon}$	Gaussian
		$\mathrm{HMM}_{\mathrm{cont}}$	$p(\mathbf{x}_0, \mathbf{x}_1, .., \mathbf{x}_T\|\boldsymbol{\theta}) = \sum_{\mathbf{z}_{0:T}} \boldsymbol{\lambda}_{\mathbf{z}_0} \prod_{t=1}^{T} \boldsymbol{\Lambda}_{\mathbf{z}_t,\mathbf{z}_{t-1}} \boldsymbol{\Gamma}_t$	$\boldsymbol{\lambda}, \boldsymbol{\Lambda}, \boldsymbol{\Gamma}$	Gaussian
Discrete	Indp.	BoF	$p(\tilde{\mathbf{x}}_t\|\boldsymbol{\theta}) = \boldsymbol{\lambda}_t$	$\boldsymbol{\lambda}$	Multinomial
	Seq.	Markov	$p(\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_T\|\boldsymbol{\theta}) = \boldsymbol{\lambda}_{\tilde{\mathbf{x}}_0} \prod_{t=1}^{T} \boldsymbol{\Lambda}_{\tilde{\mathbf{x}}_t,\tilde{\mathbf{x}}_{t-1}}$	$\boldsymbol{\lambda}, \boldsymbol{\Lambda}$	Multinomial
		$\mathrm{HMM}_{\mathrm{disc}}$	$p(\tilde{\mathbf{x}}_0, \tilde{\mathbf{x}}_1, .., \tilde{\mathbf{x}}_T\|\boldsymbol{\theta}) = \sum_{\mathbf{z}_{0:T}} \boldsymbol{\lambda}_{\mathbf{z}_0} \prod_{t=1}^{T} \boldsymbol{\Lambda}_{\mathbf{z}_t,\mathbf{z}_{t-1}} \boldsymbol{\Phi}_t$	$\boldsymbol{\lambda}, \boldsymbol{\Lambda}, \boldsymbol{\Phi}$	Multinomial

Table I

CONTINUOUS: $\mathbf{x} \in \mathbb{R}^D$,$\boldsymbol{\Lambda}_{\mathbf{z}_t,\mathbf{z}_{t-1}} = p(\mathbf{z}_t|\mathbf{z}_{t-1})$,$\boldsymbol{\Gamma}_t = p(\mathbf{x}_t|\mathbf{z}_t)$, $p\left(\mathbf{x}_t|\mathbf{z}_t{}^{(i)}\right) = \mathcal{N}\left(\mathbf{x}_t|\mu^{(i)}, \boldsymbol{\Sigma}^{(i)}\right)$, $\eta_{\mathbf{z}_0} = \mathcal{N}(\mathbf{z}_0|\mu_0, \boldsymbol{\Sigma}_0)$, $\boldsymbol{\Upsilon}_t = \mathcal{N}(\mathbf{x}_t|\mathbf{B}\mathbf{z}_t)$,$\boldsymbol{\Omega}_{\mathbf{z}_t,\mathbf{z}_{t-1}} = \mathcal{N}(\mathbf{z}_t|\mathbf{A}\mathbf{z}_{t-1})$. L IS THE NUMBER OF COMPONENTS IN THE GMM, P INDICATES THE ORDER OF THE AR MODEL, $\mathbf{A}$ AND $\mathbf{C}$ ARE THE COEFFICIENTS AND NOISE COVARIANCE IN THE AR MODEL RESPECTIVELY AND T INDICATES THE LENGTH OF THE SEQUENCE. DISCRETE: $\tilde{\mathbf{x}} \sim \mathrm{Multinomial}(\boldsymbol{\lambda})$, $\boldsymbol{\Lambda}_{\tilde{\mathbf{x}}_t,\tilde{\mathbf{x}}_{t-1}} = p(\tilde{\mathbf{x}}_t|\tilde{\mathbf{x}}_{t-1})$, $\boldsymbol{\Phi}_t = p(\tilde{\mathbf{x}}_t|\mathbf{z}_t)$, $p\left(\mathbf{x}_t|\mathbf{z}_t{}^{(i)}\right) = \mathrm{Multinomial}\left(\boldsymbol{\lambda}^{(i)}\right)$. THE BASIC MEAN REPRESENTATION IS OFTEN USED IN THE MIR FIELD IN COMBINATION WITH A SO-CALLED SQUARED EXPONENTIAL KERNEL ([61]), WHICH IS EQUIVALENT TO FORMULATING A PPK WITH A GAUSSIAN WITH THE GIVEN MEAN AND A COMMON, DIAGONAL COVARIANCE MATRIX CORRESPONDING TO THE LENGTH SCALE WHICH CAN BE FOUND BY CROSS-VALIDATION AND SPECIFICALLY USING $q = 1$ IN THE PPK.

Information Criteria and an individualized representation using cross validation.

1) Global representations: This approach is likelihood-based and penalizes the number of parameters used to estimate the model for each feature time-series. We explore the Bayesian Information Criteria (BIC) for HMM and LDS. For the AR models, there are likelihood-based criteria (e.g. AIC, BIC, etc.) and prediction-error (PFE*) approaches to determine the appropriate order of the models. This is global in the sense that the selection of parameters for the feature representation is only dependent of the individual feature time series.

2) Individual representations: Using cross validation, we can specify a feature representation individualized to each participant. The assumption is that each person listens and perceives the music differently, e.g. emphasizes different aspects and structures in the musical signal, and therefore the representation should also be individualized. We use two different types of cross-validation 1) simply sweeping across model order i.e. for AR models the temporal lag, HMM and LDS models the dimension of transition matrix 2) using the idea of information criteria but simply using crossvalidation to weigh the penalty term for the number of parameters used. The difference here is that each excerpt potentially ends up with different model orders as compared to using method 1. This in turn also examines all possible information criteria that use the same form as the AIC and BIC.

B. Kernel Function

The various track-level representations outlined above are all described in terms of a probability density as outlined in Table I, for which a natural kernel function is the Probability Product Kernel [64]. The PPK forms a common ground for comparison and is defined as,

$$k\left(p(\mathbf{x}|\boldsymbol{\theta}), p\left(\mathbf{x}|\boldsymbol{\theta}'\right)\right) = \int \left(p(\mathbf{x}|\boldsymbol{\theta})\, p\left(\mathbf{x}|\boldsymbol{\theta}'\right)\right)^q d\mathbf{x}, \quad (1)$$

where $q > 0$ is a free model parameter. The parameters of the density model, $\boldsymbol{\theta}$, obviously depend on the particular representation and are outlined in Tab.I. All the densities discussed previously result in (recursive) analytical computations, [64], [42]. It should be noted that using the PPK does not require the same length T of the sequences (the musical excerpts). For latent variable models, such as the HMM and LDS, the number of latent states in the models can also be different. The observation space, including the dimensionality D, is the only thing that has to be the same. This is convenient in the case where excerpts of different lengths should be compared.

IV. PAIRWISE KERNEL GLM

The pairwise paradigm requires an untraditional modeling approach, for which we derive a relatively simple kernel version of the Bradley-Terry-Luce model [65] for pairwise comparisons. The resulting kernel is also applicable in other kernel machines such as support vector machines.

We first collect the vector representation $\mathbf{x}$ for N audio excerpts in the set $\mathcal{X} = \{\mathbf{x}_i | i = 1, ..., N\}$, where $\mathbf{x}_i \in \mathbb{R}^D$, denotes a standard, D dimensional audio feature vector for excerpt i. In the pairwise paradigm, any two distinct excerpts with index u and v, where $\mathbf{x}_u \in \mathcal{X}$ and $\mathbf{x}_v \in \mathcal{X}$, can be compared in terms of a given aspect (such as arousal/valance). With M such comparisons, we denote the output set as $\mathcal{Y} = \{(y_m; u_m, v_m) | m = 1, ..., M\}$, where $y_m \in \{-1, +1\}$ indicates which of the two excerpts had the highest valence

(or arousal). $y_m = -1$ means that the u_m'th excerpt is picked over the v_m'th and visa versa when $y_m = 1$.

The basic assumption is that the choice, y_m, between the two distinct excerpts, u and v, can be modeled as the difference between two function values, $f(\mathbf{x}_u)$ and $f(\mathbf{x}_v)$. The function $f : \mathcal{X} \rightarrow \mathbb{R}$ hereby defines an internal, but latent absolute reference of valence (or arousal) as a function of the excerpt (represented by the audio features, $\mathbf{x}$).

Modeling such comparisons can be accomplished by the Bradley-Terry-Luce model [65], [66], here referred to more generally as the (logistic) pairwise GLM model. The choice model assumes logistically distributed noise [66] on the individual function value, and the likelihood of observing a particular choice, y_m, for a given comparison m therefore becomes

$$p\left(y_m|\mathbf{f}_m\right) \equiv \frac{1}{1+\mathrm{e}^{-y_m \cdot z_m}}, \tag{2}$$

with $z_m = f(\mathbf{x}_{u_m}) - f(\mathbf{x}_{v_m})$ and $\mathbf{f}_m = [f(\mathbf{x}_{u_m}), f(\mathbf{x}_{v_m})]^T$.

The remaining question is how the function, $f(\cdot)$, is modeled. In the following, we derive a kernel version of this model in the framework of kernel Generalized Linear Models (kGLM). We start by assuming a linear and parametric model of the form $\mathbf{f}_i = \mathbf{x}_i \mathbf{w}^\top$ and consider the likelihood defined in Eq. (2). The argument, z_m, is now redefined such that $z_m = \left(\mathbf{x}_{u_m}\mathbf{w}^\top - \mathbf{x}_{v_m}\mathbf{w}^\top\right)$. We assume that the model parameterized by $\mathbf{w}$ is the same for the first and second input, i.e. $\mathbf{x}_{u_m}$ and $\mathbf{x}_{v_m}$. This results in a projection from the audio features $\mathbf{x}$ into the dimensions of valence (or arousal) given by $\mathbf{w}$, which is the same for all excerpts. Plugging this into the likelihood function we obtain:

$$p\left(y_m|\mathbf{x}_{u_m}, \mathbf{x}_{v_m}, \mathbf{w}\right) = \frac{1}{1+\mathrm{e}^{-y_m\left(\left(\mathbf{x}_{u_m}-\mathbf{x}_{u_m}\right)\mathbf{w}^\top\right)}}. \tag{3}$$

Following a maximum likelihood approach, the effective cost function, $\psi(\cdot)$, defined as the negative log likelihood is:

$$\psi_{GLM}\left(\mathbf{w}\right) = -\sum\nolimits_{m=1}^{M} \log p\left(y_m|\mathbf{x}_{u_m}, \mathbf{x}_{v_m}, \mathbf{w}\right). \tag{4}$$

Here we assume that the likelihood factorizes over the observations, i.e. $p\left(\mathcal{Y}|\mathbf{f}\right) = \prod_{m=1}^{M} p\left(y_m|\mathbf{f}_m\right)$. Furthermore, a regularized version of the model is easily formulated as

$$\psi_{GLM-L2}\left(\mathbf{w}\right) = \psi_{GLM} + \lambda \left\|\mathbf{w}\right\|_2^2, \tag{5}$$

where the regularization parameter λ is to be found using for example cross-validation, as adopted here. This cost is still continuous and is solved with a standard optimization technique.

This basic pairwise GLM model has previously been used to model emotion in music [67]. In this work the pairwise GLM model is extended to a general regularized kernel formulation allowing for both linear and non-linear models. First, consider an unknown, non-linear map of an element $\mathbf{x} \in \mathcal{X}$ into a Hilbert space, $\mathcal{H}$, i.e., $\varphi(\mathbf{x}) : \mathcal{X} \mapsto \mathcal{H}$. Thus, the argument z_m is now given as

$$z_m = \left(\varphi\left(\mathbf{x}_{u_m}\right) - \varphi\left(\mathbf{x}_{v_m}\right)\right)\mathbf{w}^T \tag{6}$$

The *representer theorem* [68] states that the weights, $\mathbf{w}$ — despite the linear difference between mapped instances — can be written as a linear combination of the inputs such that It is easily shown that as with standard kernel logistic regression (KLR) [**?**], we can write the weights, $\mathbf{w}$, as a linear combination of the inputs in order to use the kernel trick, so with

$$\mathbf{w} = \sum\nolimits_{l=1}^{M} \alpha_l \left(\varphi\left(\mathbf{x}_{u_l}\right) - \varphi\left(\mathbf{x}_{v_l}\right)\right). \tag{7}$$

Inserting this into Eq. (6) and applying the "kernel trick" [61], i.e. exploiting that $\left\langle \varphi\left(\mathbf{x}\right)\varphi\left(\mathbf{x}'\right)\right\rangle_{\mathcal{H}} = k\left(\mathbf{x}, \mathbf{x}'\right)$, we obtain

$$\begin{aligned}
z_m &= \left(\varphi\left(\mathbf{x}_{u_m}\right) - \varphi\left(\mathbf{x}_{v_m}\right)\right)\sum_{l=1}^{M}\alpha_l\left(\varphi(\mathbf{x}_{u_l}) - \varphi(\mathbf{x}_{v_l})\right) \\
&= \sum_{m=1}^{M}\alpha_l\left(\varphi\left(\mathbf{x}_{u_m}\right) - \varphi\left(\mathbf{x}_{v_m}\right)\right)\left(\varphi(\mathbf{x}_{u_l}) - \varphi(\mathbf{x}_{v_l})\right) \\
&= \sum_{l=1}^{M}\alpha_l(\varphi\left(\mathbf{x}_{u_m}\right)\varphi(\mathbf{x}_{u_l}) - \varphi\left(\mathbf{x}_{u_m}\right)\varphi(\mathbf{x}_{v_l}) \\
&- \varphi\left(\mathbf{x}_{v_m}\right)\varphi(\mathbf{x}_{u_l}) + \varphi\left(\mathbf{x}_{v_m}\right)\varphi(\mathbf{x}_{v_l})) \\
&= \sum_{l=1}^{M}\alpha_l(k\left(\mathbf{x}_{u_m}, \mathbf{x}_{u_l}\right) - k\left(\mathbf{x}_{u_m}, \mathbf{x}_{v_l}\right) \\
&- k\left(\mathbf{x}_{v_m}, \mathbf{x}_{u_l}\right) + k\left(\mathbf{x}_{v_m}, \mathbf{x}_{v_l}\right)) \\
&= \sum_{l=1}^{M}\alpha_l k\left(\{\mathbf{x}_{u_m}, \mathbf{x}_{v_m}\}, \{\mathbf{x}_{u_l}, \mathbf{x}_{v_l}\}\right).
\end{aligned} \tag{8}$$

Thus, the pairwise kernel GLM formulation leads exactly to standard kernel GLM like [69], where the only difference is the kernel function, which is now a (valid) kernel between two sets of pairwise comparisons [5]. If the kernel between inputs is a linear kernel, we obtain the basic pairwise logistic regression presented in Eq. (3). The cost function is now defined as

$$\psi_{kGLM-L2}\left(\boldsymbol{\alpha}\right) = -\sum_{m=1}^{M}\log p\left(y_m|\boldsymbol{\alpha}, \mathbf{K}\right) + \lambda\boldsymbol{\alpha}^\top\mathbf{K}\boldsymbol{\alpha},$$

i.e. in terms of $\boldsymbol{\alpha}$, but it is of the same form as for the basic model and we can apply standard optimization techniques to find the L2 regularized solution. Predictions for unseen input pairs $\{\mathbf{x}_r, \mathbf{x}_s\}$ is easily calculated as

$$\begin{aligned}
\Delta f_{rs} &= f\left(\mathbf{x}_r\right) - f\left(\mathbf{x}_s\right) &(9)\\
&= \sum\nolimits_{m=1}^{M}\alpha_m\, k\left(\{\mathbf{x}_{u_m}, \mathbf{x}_{v_m}\}, \{\mathbf{x}_r, \mathbf{x}_s\}\right). &(10)
\end{aligned}$$

Thus, as seen from Eq. (8), predictions exist naturally only as delta predictions. however. it is easy to obtain a "true" latent (arbitrary scale) function for a single output by aggregating all the delta predictions. To evaluate the different feature representations, two datasets are used. The first dataset consists of $N_{\text{IMM}} = 20$ excerpts and is described in [71]. It comprises all $M_{\text{IMM}} = 190$ unique pairwise comparisons of 20 different 15 second excerpts, chosen from the USPOP2002[1] dataset. 13 participants (3 female, 10 male) were compared on both the

[5] In the Gaussian Process setting this kernel is also known as the Pairwise Judgment kernel [70].

[1] http://labrosa.ee.columbia.edu/projects/musicsim/uspop2002.html

dimensions of valence and arousal. The second dataset [72] consists of $M_{\text{YANG}} = 7752$ pairwise comparisons made by multiple annotators on different parts of the $N_{\text{YANG}} = 1240$ different Chinese 30-second-long excerpts, on the dimension of valence.

Feature	Description	Dimension
Mel-frequency cepstral coefficients (MFCC)[6]	The discrete cosine transform of the log-transformed short-time power spectrum on the logarithmic mel-scale.	20
Chromagram [73]	The short-time energy spectrum is computed and summed appropriately to form each pitch class.	12
Loudness [74]	Loudness is the energy in each critical band.	24
Echonest *Timbre*[7]	Proprietary features to describe timbre.	12
Echonest *Pitch*[??]	Proprietary chroma-like features.	12

Table II
ACOUSTIC FEATURES USED FOR EMOTION PREDICTION.

A. Performance Evaluation

In order to evaluate the performance of the proposed representation of the multivariate feature time series, we compute learning curves. We use the so-called Leave-One-Excerpt-Out cross validation, which ensures that all comparisons with a given excerpt are left out in each fold [67]. Furthermore a 'win'-based baseline ($Base_{low}$) as suggested in [71] is used. This baseline represents a model with no information from features, i.e. testing against this baseline tests whether information is found in the features for predicting expressed emotion in music represented by the pairwise comparisons. We use the McNemar paired test between each model and the baseline, with the *Null* hypothesis that the two models are the same, if $p < 0.05$ then the models can be rejected as equal on a 5% significance level.

V. RESULTS

We consider the pairwise classification error on the two outlined datasets with the L2 regularized pairwise kernel GLM model, and the outlined pairwise kernel function combined with the PPK kernel (with q=1/2). For the *YANG* dataset, a global regularization parameter λ was estimated using 5-fold cross validation. The 14 different track-level representations are evaluated on the 5 different features extracted from the two datasets. The quantization of the multivariate time series, i.e. the vector quantization, was performed using a standard online K-means algorithm, namely sofia K-means [75] with random initialization and a standard Euclidean metric. To prevent overfitting, the codebook was estimated for each LOEO fold on the *IMM* dataset and for the *YANG* dataset the codebooks were estimated on the entire dataset. The estimations where chosen as the best representation out of 10 repetitions. The codebook sizes for the temporal models were 8, 16, 24 and 32 audiowords and for the VQ models 256, 512 and 1024 were tested. For the continuous emitting HMMs, 2 to 5 states were chosen, with a single Gaussian for each state. Introducing a GMM for each state did not show any performance improvements, and using more states only made the model estimation more difficult due to the small number of samples in these feature time series. Similarly, the LDS/DT models showed that only a low dimensionality of the transition matrix was possible. Hyperparameters and kernel parameters were estimated individually for each participant in the *IMM* dataset, whereas for the *YANG* dataset global parameters were estimated.

We present the results comparing the two different domains of feature representations namely a continuous and discretized representation.

A. Continuous

Comparing the performance of the kGLM-L2 model predicting pairwise comparisons using the 5 different features on the *YANG* dataset, on average across feature representations we see that the MFCC, Loudness and Echonest Pitch features are the best-performing, while Chroma and Echonest Timbre perform rather poorly. The traditional approach of taking the mean across the entire multivariate time-series of the 30-second excerpts is the worst-performing representation. Increasing complexity and using a single Gaussian with diagonal covariance improves performance for all features, and using a full covariance further improves the representation. Introducing additional Gaussians using a GMM is the best-performing, non-temporal representation and for the Chroma features is the best method of representing the features for this specific task. Introducing temporal coding using the HMM with full covariance Gaussian emissions does not improve performance for most features except for Echonest Pitch features, where an improvement of 1.9% compared to a full Gaussian is observed. The AR models, as previously shown in [7], perform very well in coding MFCC features, likewise for Loudness and Echonest Timbre. For loudness, a diagonal model with order of p=9 is the best performing, whereas for MFCCs the VAR model performs best; again with rather high order of p=4. Adding an extra dimension of complexity with the latent dimensions of the LDS/DT model does not seem to improve the feature representation, regardless of how the complexity of the model is chosen. The AIC and BIC for the Diagonal AR models do not perform well across the different features in selecting an order that is useful as feature representation for emotion prediction, whereas the FPE for the VAR model seems to be a good method.

Using the continuous representation, the AR models again have the best predictive performance, with the best performance obtained for all but echonest pitch features. The strategy of finding the best order is again using cross validation. The AIC and BIC perform rather poorly compared to CV and completely fail for the echonest features due to selecting too high orders, making the computation of the PPK improper. The order selected is rather low (p=2-5) as compared to the order selected for the valence data on both the *YANG* and *IMM* datasets (p=6-10). It seems that the more complex LDS/DT model does not perform that well for the arousal data for any of the features as compared to the related AR models.

Using the HMM models with continuous emissions shows rather poor predictive performance, here included using both

Obs.	Time	States	Models	Features				
				MFCC	Chroma	Loudness	Timbre	Pitch
Continuous	Indp.	Observed	Mean	0.256	0.332	0.283	0.311*	0.269
			$\mathcal{N}(\mathbf{x}\|\mu,\sigma)$	0.254	0.295*	0.269	0.307*	0.272
			$\mathcal{N}(\mathbf{x}\|\mu,\mathbf{\Sigma})$	0.239	0.280	0.242	0.276	0.250
			$\text{GMM}_{diag,BIC}$	0.238	0.252	0.238	0.270	0.260
			$\text{GMM}_{full,BIC}$	0.229	**0.250**	0.235	0.264	0.247
	Temp.	Observed	$\text{AR}_{dar,p=7}$	0.245	0.264	0.238	0.263	0.239
			$\text{AR}_{dar,p=8}$	0.242	0.273	0.235	0.259	0.238
			$\text{AR}_{dar,p=9}$	0.241	-	**0.234**	0.257	0.237
			$\text{AR}_{dar,p=10}$	0.239	-	0.342	0.258	0.237
			$\text{AR}_{dar,AIC}$	0.306	0.273	0.238	0.300	0.270
			$\text{AR}_{dar,BIC}$	0.250	0.285	0.256	0.266	0.243
			$\text{AR}_{var,p=1}$	0.253	0.296*	0.254	0.277	0.254
			$\text{AR}_{var,p=2}$	0.233	0.273	0.239	**0.257**	0.245
			$\text{AR}_{var,p=3}$	0.223	0.271	1.000	-	-
			$\text{AR}_{var,p=4}$	**0.221**	0.274	1.000	-	-
			$\text{AR}_{var,FPE}$	0.223	0.271	0.239	0.277	0.254
		Latent	$\text{LDS}_{full,s=1}$	0.265	0.277	0.264	0.269	0.257
			$\text{LDS}_{full,s=2}$	0.262	0.273	0.261	0.257	0.255
			$\text{LDS}_{full,s=3}$	0.269	0.276	0.265	0.259	0.254
			$\text{LDS}_{full,BIC}$	0.273	0.274	0.269	0.264	0.258
			$\text{HMM}_{full,s=2}$	0.232	0.255	0.240	0.258	**0.231**
			$\text{HMM}_{full,s=3}$	0.233	0.254	0.241	0.257	0.240
			$\text{HMM}_{full,s=4}$	0.233	0.251	0.241	0.262	0.249
			$\text{HMM}_{full,s=5}$	0.232	0.250	0.243	0.264	0.255
Discrete	Indp.	Observed	$\text{VQ}_{p=256}$	0.268	0.329	0.282	0.327	0.267
			$\text{VQ}_{p=512}$	0.266	0.313	0.280	0.319	0.260
			$\text{VQ}_{p=1024}$	0.264	0.308*	0.272	0.308*	0.257
	Temp.	Observed	$\text{Markov}_{p=8}$	0.260	0.305	0.264	0.260	0.245
			$\text{Markov}_{p=16}$	0.236	0.258	0.257	*0.259*	0.236
			$\text{Markov}_{p=24}$	0.233	*0.256*	0.244	0.260	*0.235*
			$\text{Markov}_{p=32}$	*0.233*	0.257	*0.237*	0.263	0.238
		Latent	$\text{HMM}_{s=2,p=8}$	0.268	0.309*	0.279	0.282	0.248
			$\text{HMM}_{s=3,p=8}$	0.264	0.310*	0.275	0.270	0.246
			$\text{HMM}_{s=4,p=8}$	0.248	0.293	0.262	0.263	0.252
			$\text{HMM}_{BIC,p=8}$	0.268	0.305	0.291	0.275	0.248
			$\text{HMM}_{s=2,p=16}$	0.254	0.272	0.254	0.261	0.238
			$\text{HMM}_{s=3,p=16}$	0.256	0.270	0.248	0.262	0.244
			$\text{HMM}_{s=4,p=16}$	0.255	0.269	0.246	0.266	0.246
			$\text{HMM}_{BIC,p=16}$	0.268	0.264	0.277	0.261	0.243
			$\text{HMM}_{s=2,p=24}$	0.262	0.265	0.252	0.271	0.253
			$\text{HMM}_{s=3,p=24}$	0.261	0.261	0.251	0.264	0.246
			$\text{HMM}_{s=4,p=24}$	0.262	0.271	0.257	0.265	0.245
			$\text{HMM}_{BIC,p=24}$	0.261	0.261	0.256	0.264	0.244
			$\text{HMM}_{s=2,p=32}$	0.259	0.266	0.250	0.260	0.240
			$\text{HMM}_{s=3,p=32}$	0.260	0.265	0.256	0.260	0.262
			$\text{HMM}_{s=4,p=32}$	0.258	0.268	0.262	0.264	0.246
			$\text{HMM}_{BIC,p=32}$	0.259	0.264	0.254	0.263	0.248
			Baseline	0.262	0.262	0.262	0.262	0.262

Table III

RESULTS OF THE KGLM-L2 MODEL EVALUATING DIFFERENT FEATURE AND FEATURE REPRESENTATION USING A 10-FOLD CROSS VALIDATION ERROR RATE PERFORMED ON THE *YANG* DATASET EVALUATING THE DIMENSION OF **VALENCE**. RESULTS IN **BOLD** INDICATE THE BEST PERFORMING FEATURE REPRESENTATION FOR THAT PARTICULAR FEATURE AND RESULTS IN *italic* INDICATE THE BEST PERFORMING FEATURE REPRESENTATION FOR EITHER CONTINUOUS OR DISCRETE OBSERVATION SPACE.

a diagonal and full covariance emission distribution. Multiple attempts were made to find different implementations and use multiple initializations to find suitable models to represent the feature, but with no luck. Internally, attempts with higher orders (s¿5) and using GMM emissions (p¿1) for each latent state were made, but with little difference. Here we do see some difference between using a diagonal emission distribution, whereas for the *YANG* dataset no difference was observed.

The valence data for the *IMM* dataset on figure V again shows that the MFCC features perform well as previously shown [7]. The AR models again show a great performance improvement as compared to any of the independent models across all features except for the echonest pitch features. Surprisingly, many of the non-temporal representations perform very poorly, being non-significant from baseline in many cases. We do see that adding the extra latent dimensions using the LDS/DT model is beneficial as feature-representation for chroma and echonest timbre features, when selecting states using CV. Using a full correlation between each feature dimension in the VAR model seems to perform poorly compared to only using a diagonal model in the DAR.

B. Discretized

Looking at the discretized features, the three different independent models are the worst performing of the feature representations. Surprisingly, the VQ performs equally poorly compared to simply taking the average across the features.

Adding some temporal coding of the discretized features using the Markov model yields a great improvement in the performance. Across all features used in this work, it is the best performing feature representation for discretized data. For the HMM there is a decrease in performance and it seems the structures this more complex model finds are not suitable to predict the valence annotations of the *YANG* dataset.

For the discretized acoustical features, the Markov model shows a slight improvement as compared to the Vector Quantization data across all features for the *IMM* arousal dataset. For the MFCC, Chroma, and echonest timbre feature there is a significant difference between best performing VQ and Markov representation, whereas for loudness there is no significant difference. Increasing the memory in the feature representation using the HMM models of the discretized data we specially see a performance improvement for the Chroma and echonest pitch features as compared to the Markov representation, where in the echonest pitch was not a significant improvement compared to a simple VQ representation. The selection of latent states shows again that the information-criteria approach fails compared to the two cross-validation strategies. However comparing the $HMM_{WIC,p=16}$ and $HMM_{CV,p=16}$ there is no significant difference.

In the discretized feature case, the simple-independent VQ representation performs surprisingly badly, not just in the MFCC case, but all features. Also here where we see the biggest and consistent performance improvements when increasing temporal complexity in the feature representation. For the MFCC features going from VQ to Markov means an absolute improvement is obtained of 3.34% and relative of (11.95%) and for Chroma of 2.55% (8.57%), Loudness of 4.25% (14.68%) and echonest timbre of 2.06% (7.14%). Increasing the complexity further from a Markov to an HMM representation the performance further increases for MFCC features of 1.34% (5.43%), Chroma of 3.95% (14.50%), Loudness of 1.34% (5.41%), echonest timbre of 1.28% (4.75%) and echonest pitch of 1.94% (7.78%)

VI. Discussion

In essence, we are looking for a way of representing an entire track based on the simple features extracted. That is, we are trying to find generative models that can capture meaningful information coded in the features specifically for coding aspects related to the emotions expressed in music. In this case, we compare single features with single representations, finding which single representation is most suitable for each feature. Since these are unsupervised methods, we perform an explorative approach in finding which feature and feature representation combination is most suitable. The advantage of using this framework is that we can use any generative model for feature representation and using such a representation replace the entire feature time series by the model, since the distances between tracks are now between the models trained on each of the tracks and not directly on the features. This provides a significant reduction in the number of parameters to store for each track. Furthermore, it allows us to code different temporal structures for each feature and potentially combine features extracted on different time scales.

The five features each represent aspects of music which could explain the emotions expressed in music. The MFCC and Echonest timbre features are said to capture timbre, whereas the Chroma and Echonest pitch are both of tonal character and the loudness being of psychoacoutic origin.

A. Discretized

When discretizing features such as the Echonest pitch and Chroma features, using k-means we can analyze the codewords. The first codewords using $p = 8$ and $p = 16$ are essentially single and double tones. As the number of codewords ($p > 16$) increase, we see more and more complex chords. This means that when using a VQ model ($p > 255$) it thus codes which keys/chords are present in the track. Coding the tonal keywords using a Markov or HMM model essentially produces a probabilistic key and chord-transition representation. The predictive performance difference between coding only the presence of keys/chords and coding transitions can clearly be seen across all datasets used. On the arousal data, an increase of 3.8% and valence of 6.5% for the *IMM* dataset and 5.2% on the *YANG* dataset. [8] The echonest pitch feature, using a severely reduced temporal resolution, does not show the same improvement using the smaller 15-second excerpt of the *IMM* dataset, whereas for the *YANG* dataset an improvement of 2.2% is obtained. Across all datasets we observed that this reduced temporal resolution in the echonest features made the estimation of representations rather hard and did not aid in gaining any more detailed insight into the temporal dynamics. Discretizing the Loudness features captures different energy patterns across the critical bands used in the loudness model. Using the Markov and HMM produces a dynamical loudness representation. Only using the presence of loudness patterns shows rather poor predictive performance of the valence data both *IMM* and *YANG* dataset but coding the transitions with Markov and HMM improves the performance significantly. For arousal data, however, this does not seem to be the case, which is something that should be looked further into. When discretizing MFCC and Echonest timbre features, the codewords can be somewhat hard to interpret. We do however see the same pattern that simply using a VQ is performing poorly as compared to the Markov and HMM.

Using the smaller excerpts of 15 seconds in the *IMM* dataset seems to favor the HMMs. Naturally more memory is present in the HMMs as compared to the Markov models, thus enabling the coding of more complex temporal structures, which is essential for coding the valence dimension across all the features. Thus potentially finding hidden structures in the features not coded in each frame of the features but, by their longer term temporal structures, captured by the models.

B. Continuous

We see the same trend with the continuous observations, i.e. including temporal information significantly increases predictive performance. This is the case for all features used to

[8] To compare the difference in the number of keywords used for the VQ models and Markov and HMMs, the VQ representation was used for the same codewords as the Markov and HMMs and performed very poorly.

Obs.	Time	States	Models	Features				
				MFCC	Chroma	Loudness	Timbre	Pitch
Continuous	Indp.	Observed	Mean	0.203	0.282*	0.219	0.215	0.228
			$\mathcal{N}(\mathbf{x}\|\mu,\sigma)$	0.188	0.228	0.202	0.215	0.214
			$\mathcal{N}(\mathbf{x}\|\mu,\mathbf{\Sigma})$	0.205	0.244	0.215	0.240	0.228
			$\text{GMM}_{diag,BIC}$	0.249	0.256	0.224	0.260	0.242
			$\text{GMM}_{full,BIC}$	0.213	0.264	0.224	0.272*	0.272*
			$\text{GMM}_{diag,CV}$	0.189	0.222	0.202	0.212	*0.213*
			$\text{GMM}_{full,CV}$	0.205	0.244	0.214	0.239	0.228
	Temp.	Observed	$\text{AR}_{dar,p=2}$	0.198	0.235	*0.179*	0.214	0.232
			$\text{AR}_{dar,p=3}$	0.195	0.251	0.181	0.219	0.251
			$\text{AR}_{dar,p=4}$	0.188	0.256	0.196	0.239	0.261
			$\text{AR}_{dar,CV}$	**0.175**	*0.214*	**0.172**	**0.202**	0.213
			$\text{AR}_{dar,AIC}$	0.200	0.265*	0.595	-	-
			$\text{AR}_{dar,BIC}$	0.195	0.259	0.181	-	-
			$\text{AR}_{var,p=1}$	0.184	0.264	0.182	0.244	0.260
			$\text{AR}_{var,p=2}$	0.201	0.270*	0.195	1.000	1.000
			$\text{AR}_{var,CV}$	0.181	0.259	0.180	0.244	0.260
		Latent	$\text{LDS}_{full,WIC}$	0.241	0.250	0.210	0.233	0.254
			$\text{LDS}_{full,BIC}$	0.276	0.265*	0.254	0.244	0.261*
			$\text{LDS}_{full,CV}$	0.232	0.218	0.204	0.224	0.237
			$\text{HMM}_{full,s=2}$	0.258	0.251	0.263	0.262	0.270*
			$\text{HMM}_{full,s=3}$	0.266*	0.255	0.261	0.265	0.269*
			$\text{HMM}_{full,s=4}$	0.264*	0.250	0.265*	0.273*	0.267*
			$\text{HMM}_{full,s=5}$	0.256	0.241	0.267*	0.273	0.272*
			$\text{HMM}_{diag,s=2}$	0.239	0.240	0.255	0.247	0.265*
			$\text{HMM}_{diag,s=3}$	0.254	0.243	0.267*	0.262	0.281
			$\text{HMM}_{diag,s=4}$	0.261	0.249	0.273	0.257	0.269*
			$\text{HMM}_{diag,s=5}$	0.260	0.244	0.270*	0.257	0.287
Discrete	Indp.	Observed	$\text{VQ}_{p=256}$	0.188	0.251	0.185	0.232	0.201
			$\text{VQ}_{p=512}$	0.188	0.243	*0.176*	0.231	0.213
			$\text{VQ}_{p=1024}$	0.189	0.244	0.179	0.246	0.210
	Temp.	Observed	$\text{Markov}_{p=8}$	0.190	0.258	0.177	0.228	0.230
			$\text{Markov}_{p=16}$	*0.178*	0.242	0.178	0.247	0.216
			$\text{Markov}_{p=24}$	0.195	0.231	0.191	0.224	0.220
			$\text{Markov}_{p=32}$	0.197	0.240	0.199	0.243	0.236
		Latent	$\text{HMM}_{p=8,CV}$	0.193	0.225	0.182	*0.217*	0.229
			$\text{HMM}_{p=8,BIC}$	0.247	0.267*	0.207	0.235	0.243
			$\text{HMM}_{p=16,CV}$	0.196	**0.207**	0.185	0.229	0.234
			$\text{HMM}_{p=16,BIC}$	0.218	0.236	0.203	0.240	0.257
			$\text{HMM}_{p=24,CV}$	0.214	0.220	0.205	0.234	0.203
			$\text{HMM}_{p=24,BIC}$	0.212	0.245	0.238	0.250	0.238
			$\text{HMM}_{p=32,CV}$	0.199	0.228	0.191	0.229	**0.200**
			$\text{HMM}_{p=32,BIC}$	0.229	0.260	0.223	0.243	0.254
			Baseline	0.269	0.269	0.269	0.269	0.269

Table IV
RESULTS OF THE KGLM-L2 MODEL EVALUATING DIFFERENT FEATURE AND FEATURE REPRESENTATION USING A 10-FOLD CROSS VALIDATION ERROR RATE PERFORMED ON THE *IMM* DATASET EVALUATING THE DIMENSION OF **AROUSAL**

predict the *YANG* dataset, except for the chroma feature. Using a 10-fold CV scheme as compared to [7] makes the results non-comparable, but the VAR model is still the best feature representation for the MFCC features. Surprisingly, a non-temporal representation of the Chroma features performs well, essentially only coding key/chord presence in the entire track. The AR model is a very fast and easy method of obtaining a track representation, however, choosing order is tricky and across all datasets the best approach seems to be cross validation. The HMMs with continuous emission are rather hard to estimate and show only slight or no improvements compared to the other continuous feature representation. In the longer sequence in the *YANG* dataset, the LDS/DT models do not show any improvement as compared to the simpler AR models. The same applies for the arousal data in the *IMM* dataset, but for the valence dataset we see a rather large improvement across all features used. This shows there is no clear-cut case in disregarding any feature representation.

C. Model selection

A challenge across latent variable models like GMM, HMM and LDS and for observed-state models like AR models, is model selection. We have investigated two different situations, namely a personalized case, where representations are fitted specifically for each subject, and a static case, where no information is present about each user in the *YANG* case.

Individual representations

The idea of using individualized feature representation specific for each user works very well for the *IMM* dataset for both valence and arousal. Although it is a rather small dataset, we do see that using individualized model orders of the AR models produces a dramatic performance gain and should be further investigated on larger datasets where the details of each user is known. The same observation goes for the GMM, LDS and HMM models, that comparing the use of global representations shows rather poor performance compared to finding individual model orders. The downside is that more resources should be invested in finding these model orders.

Obs.	Time	States	Models	Features				
				MFCC	Chroma	Loudness	Timbre	Pitch
Continuous	Indp.	Observed	Mean	0.272	0.446	0.270	0.301	*0.251*
			$\mathcal{N}(\mathbf{x}\|\mu,\sigma)$	0.287*	0.276*	0.282*	0.301	0.270
			$\mathcal{N}(\mathbf{x}\|\mu,\mathbf{\Sigma})$	0.255	0.277*	0.267	0.294	0.254
			$\text{GMM}_{diag,BIC}$	0.262	0.290*	0.269	0.291	0.283*
			$\text{GMM}_{full,BIC}$	0.254	0.283*	0.277*	0.283*	0.287*
			$\text{GMM}_{diag,CV}$	0.252	0.268	0.262	0.274	0.264
			$\text{GMM}_{full,CV}$	0.257	0.272	0.269	0.277	0.257
	Temp.	Observed	$\text{AR}_{dar,p=8}$	0.244	0.333	0.255	0.284*	0.280*
			$\text{AR}_{dar,p=9}$	0.237	0.380	0.259	0.284*	0.284*
			$\text{AR}_{dar,p=10}$	0.234	-	0.263	0.284*	0.285*
			$\text{AR}_{dar,CV}$	**0.225**	0.251	*0.240*	0.262	0.256
			$\text{AR}_{dar,AIC}$	0.237	0.281*	0.512	-	-
			$\text{AR}_{dar,BIC}$	0.308	0.273	0.263	-	-
			$\text{AR}_{var,p=1}$	0.275*	0.283*	0.280*	0.286*	0.269
			$\text{AR}_{var,p=2}$	0.260	0.287*	0.261	1.000	1.000
			$\text{AR}_{var,CV}$	0.246	0.275*	0.250	0.286*	0.269
		Latent	$\text{LDS}_{full,WIC}$	0.241	0.265	0.249	0.271	0.279*
			$\text{LDS}_{full,BIC}$	0.286*	0.265	0.277*	0.273	0.283*
			$\text{LDS}_{full,CV}$	0.229	*0.247*	0.248	**0.253**	0.268
			$\text{HMM}_{full,s=2}$	0.275	0.282*	0.282*	0.283*	0.281*
			$\text{HMM}_{full,s=3}$	0.284*	0.279	0.286*	0.289	0.285*
			$\text{HMM}_{full,s=4}$	0.283*	0.272	0.283*	0.289	0.284*
			$\text{HMM}_{full,s=5}$	0.280*	0.279*	0.284*	0.283*	0.284*
			$\text{HMM}_{diag,s=2}$	0.282*	0.265	0.277	0.279	0.262
			$\text{HMM}_{diag,s=3}$	0.281*	0.275*	0.280*	0.283*	0.286*
			$\text{HMM}_{diag,s=4}$	0.276*	0.283*	0.283*	0.283*	0.280*
			$\text{HMM}_{diag,s=5}$	0.286*	0.277*	0.282*	0.285*	0.292
Discrete	Indp.	Observed	$\text{VQ}_{p=256}$	0.286*	0.321	0.313	0.315	0.258
			$\text{VQ}_{p=512}$	0.282*	0.308	0.304	0.289*	0.255
			$\text{VQ}_{p=1024}$	0.280*	0.298	0.290*	0.291	0.247
	Temp.	Observed	$\text{Markov}_{p=8}$	0.252	0.277*	0.256	0.268	0.261
			$\text{Markov}_{p=16}$	0.248	0.279*	0.247	0.272	0.252
			$\text{Markov}_{p=24}$	0.246	0.272	0.253	0.272	0.262
			$\text{Markov}_{p=32}$	0.254	0.272	0.256	0.277*	0.249
		Latent	$\text{HMM}_{p=8,CV}$	0.242	0.251	0.247	0.260	0.253
			$\text{HMM}_{p=8,BIC}$	0.253	0.275*	0.283*	0.277*	0.283*
			$\text{HMM}_{p=16,CV}$	*0.233*	0.248	0.241	*0.256*	0.240
			$\text{HMM}_{p=16,BIC}$	0.267	0.273	0.269	0.279*	0.274
			$\text{HMM}_{p=24,CV}$	0.244	**0.233**	0.238	0.260	0.234
			$\text{HMM}_{p=24,BIC}$	0.261	0.275	0.271	0.281*	0.250
			$\text{HMM}_{p=32,CV}$	0.255	0.235	**0.234**	0.257	**0.230**
			$\text{HMM}_{p=32,BIC}$	0.266	0.275	0.263	0.283*	0.251
			Baseline	0.285	0.285	0.285	0.285	0.285

Table V
RESULTS OF THE KGLM-L2 MODEL EVALUATING DIFFERENT FEATURE AND FEATURE REPRESENTATION USING A 20-FOLD CROSS VALIDATION ERROR RATE PERFORMED ON THE *IMM* DATASET EVALUATING THE DIMENSION OF **VALENCE**

Comparing the weighted information criteria with the more simple CV approach showed that the information criteria approach is not the way to go for feature representation in any form evaluated on this specific dataset.

Global representations

For the GMM model, using the BIC when information is present about each user's annotations seems like a rather poor approach, but on the *YANG* dataset, performance is good. For the *IMM* dataset all criteria were not a great success, as the FPE failed to find proper representation and in some cases it was similar for the AIC and BIC case. For the LDS/DT models, CV was clearly the best performing strategy, using the same order for all excerpts or using BIC performed poorly. The same story with the HMMs across all datasets showed that using BIC is not appropriate for feature representation given these datasets.

D. Future work

We have here worked with using one single feature and feature representation combinations, however, this is potentially a simplified view of music. Different musical features and structures most likely can explain what emotions are expressed in music. Thus combining both features and feature representations would be an obvious extension to the existing approach. Given the framework presented here, using generative models and the PPK, this could be achieved using Multiple Kernel Learning - essentially learning optimal feature and feature representation combinations. Another extension would be to use more rich representations such as spectrograms, and still use the same approach as presented here.

VII. CONCLUSION

In this work, we provided a general review of current audio and feature representations focusing on the temporal aspect of modeling music. We identified and presented a

general probabilistic approach for evaluating various track-level representations for modeling and predicting higher-level aspects of music, such as genre, tag, emotion and similarity, focusing on the benefit of modeling temporal aspects of music. With the aim to do a thorough comparison between many different temporal representations, we focused on one of these aspects; namely emotion expressed in music. Here we considered datasets based on robust, pairwise paradigms for which we extended a particular kernel-based model forming a common ground for comparing different track-level representations of music using the probability product kernel. A wide range of generative models for track-level representations was considered on two datasets, focusing on evaluating using both continuous and discretized observations. Modeling the valence and arousal dimensions of expressed emotion showed a significant gain in applying temporal modeling on both the datasets included in this work. In conclusion, we have found evidence for the hypothesis that a statistically significant gain is obtained in predictive performance by representing the temporal aspect of music for emotion prediction using five different features.

REFERENCES

[1] Y.-H. Yang and H. H. Chen, "Machine recognition of music emotion: A review," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 3, no. 3, p. 40, 2012.

[2] F. Pachet and J.-J. Aucouturier, "Improving timbre similarity: How high is the sky?" *Journal of negative results in speech and audio sciences*, vol. 1, no. 1, pp. 1–13, 2004.

[3] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney, "Content-based music information retrieval: Current directions and future challenges," *Proceedings of the IEEE*, vol. 96, no. 4, pp. 668–696, 2008.

[4] R. Panda, R. Malheiro, B. Rocha, A. Oliveira, and R. Paiva, "Multi-modal music emotion recognition: A new dataset, methodology and comparative analysis," *Proc. CMMR*, 2013.

[5] B. L. Sturm, "The state of the art ten years after a state of the art: Future research in music information retrieval," *Journal of New Music Research*, vol. 43, no. 2, pp. 147–172, 2014.

[6] A. Meng, P. Ahrendt, J. Larsen, and L. K. Hansen, "Temporal feature integration for music genre classification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1654–1664, 2007.

[7] J. Madsen, B. S. Jensen, and J. Larsen, "Modeling temporal structure in music for emotion prediction using pairwise comparisons," in *15th International Conference on Music Information Retrieval (ISMIR)*, 2014.

[8] M. Henaff, K. Jarrett, K. Kavukcuoglu, and Y. LeCun, "Unsupervised learning of sparse features for scalable audio classification." in *ISMIR*, 2011, pp. 681–686.

[9] L. Su, C.-C. Yeh, J.-Y. Liu, J.-C. Wang, and Y.-H. Yang, "A systematic evaluation of the bag-of-frames representation for music information retrieval," *Multimedia, IEEE Transactions on*, vol. 16, no. 5, pp. 1188–1200, Aug 2014.

[10] C.-C. M. Yeh and Y.-H. Yang, "Supervised dictionary learning for music genre classification," in *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval*. ACM, 2012, p. 55.

[11] P. Hamel, S. Lemieux, Y. Bengio, and D. Eck, "Temporal pooling and multiscale learning for automatic annotation and ranking of audio," in *12th International Conference on Music Information Retrieval (ISMIR)*, 2011.

[12] E. M. Schmidt and Y. E. Kim, "Modeling musical emotion dynamics with conditional random fields," in *12th International Conference on Music Information Retrieval (ISMIR)*, 2011.

[13] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[14] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl, "Aggregate features and adaboost for music classification," *Machine Learning*, vol. 65, no. 2-3, pp. 473–484, 2006.

[15] E. Pampalk, A. Rauber, and D. Merkl, "Content-based organization and visualization of music archives," in *Proceedings of the tenth ACM international conference on Multimedia*. ACM, 2002, pp. 570–579.

[16] P. Hamel, Y. Bengio, and D. Eck, "Building musically-relevant audio features through multiple timescale representations." in *ISMIR*, 2012, pp. 553–558.

[17] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, "A regression approach to music emotion recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 2, pp. 448–457, 2008.

[18] M. Barthet, G. Fazekas, and M. Sandler, "Music emotion recognition: From content-to context-based models," in *From Sounds to Music and Emotions*. Springer, 2013, pp. 228–252.

[19] R. Grosse, R. Raina, H. Kwong, and A. Y. Ng, "Shift-invariance sparse coding for audio classification," *arXiv preprint arXiv:1206.5241*, 2012.

[20] J. Nam, J. Herrera, M. Slaney, and J. O. Smith, "Learning sparse feature representations for music annotation and retrieval," in *ISMIR*, 2012, pp. 565–570.

[21] C.-C. M. Yeh and Y.-H. Yang, "Towards a more efficient sparse coding based audio-word feature extraction system," in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*. IEEE, 2013, pp. 1–7.

[22] T. Blumensath and M. Davies, "Sparse and shift-invariant representations of music," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 1, pp. 50–57, 2006.

[23] R. F. Lyon, M. Rehn, S. Bengio, T. C. Walters, and G. Chechik, "Sound retrieval and ranking using sparse auditory representations," *Neural computation*, vol. 22, no. 9, pp. 2390–2416, 2010.

[24] H. Lee, P. Pham, Y. Largman, and A. Y. Ng, "Unsupervised feature learning for audio classification using convolutional deep belief networks," in *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, Eds. Curran Associates, Inc., 2009, pp. 1096–1104.

[25] P. Hamel and D. Eck, "Learning features from music audio with deep belief networks," in *ISMIR*. Utrecht, The Netherlands, 2010, pp. 339–344.

[26] E. J. Humphrey, J. P. Bello, and Y. LeCun, "Feature learning and deep architectures: new directions for music informatics," *Journal of Intelligent Information Systems*, vol. 41, no. 3, pp. 461–481, 2013.

[27] S. Dieleman, P. Brakel, and B. Schrauwen, "Audio-based music classification with a pretrained convolutional network," in *12th International Society for Music Information Retrieval Conference (ISMIR-2011)*. University of Miami, 2011, pp. 669–674.

[28] S. Tran, D. Wolff, T. Weyde, and A. d. Garcez, "Feature preprocessing with restricted boltzmann machines for music similarity learning," in *Audio Engineering Society Conference: 53rd International Conference: Semantic Audio*. Audio Engineering Society, 2014.

[29] E. M. Schmidt, J. Scott, and Y. E. Kim, "Feature learning in dynamic environments: Modeling the acoustic structure of musical emotion," in *13th International Conference on Music Information Retrieval (ISMIR)*, 2012.

[30] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6645–6649.

[31] J. Arenas-García, K. B. Petersen, and L. K. Hansen, "Sparse kernel orthonormalized pls for feature extraction in large datasets," *Nips 2006*, 2006.

[32] S. Dieleman and B. Schrauwen, "Multiscale approaches to music audio feature learning." in *ISMIR*, 2013, pp. 3–8.

[33] S. Scholler and H. Purwins, "Sparse approximations for drum sound classification," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 5, no. 5, pp. 933–940, 2011.

[34] K. K. Chang, C. S. Iliopoulos, and J.-S. R. Jang, "Music genre classification via compressive sampling," *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010*, pp. 387–392, 2010.

[35] Y. Vaizman, B. McFee, and G. R. G. Lanckriet, "Codebook based audio feature representation for music information retrieval," *CoRR*, vol. abs/1312.5457, 2013.

[36] B. McFee, L. Barrington, and G. Lanckriet, "Learning content similarity for music recommendation," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 8, pp. 2207–2218, 2012.

[37] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "Music classification via the bag-of-features approach," *Pattern Recognition Letters*, vol. 32, no. 14, pp. 1768–1777, 2011.

[38] K. Ellis, E. Coviello, and G. R. Lanckriet, "Semantic annotation and retrieval of music using a bag of systems representation." in *ISMIR*,

2011, pp. 723–728.
[39] K. Seyerlehner, G. Widmer, and P. Knees, "Frame level audio similarity-a codebook approach," in *Proceedings of the 11th International Conference on Digital Audio Effects (DAFx-08), Espoo, Finland*, 2008.
[40] J. Wülfing and M. Riedmiller, "Unsupervised learning of local features for music classification." in *ISMIR*, 2012, pp. 139–144.
[41] C. Joder, S. Essid, and G. Richard, "Temporal integration for audio classification with application to musical instrument classification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 1, pp. 174–186, 2009.
[42] A. Meng and J. Shawe-Taylor, "An investigation of feature models for music genre classification using the support vector classifier," in *International Conference on Music Information Retrieval*, 2005, pp. 604–609.
[43] P.-S. Huang, J. Yang, M. Hasegawa-Johnson, F. Liang, and T. S. Huang, "Pooling robust shift-invariant sparse representations of acoustic signals." in *INTERSPEECH*, 2012.
[44] P. Foster, M. Mauch, and S. Dixon, "Sequential complexity as a descriptor for musical similarity," *arXiv preprint arXiv:1402.6926*, 2014.
[45] N. Scaringella and G. Zoia, "On the modeling of time information for automatic genre recognition systems in audio signals," in *6th International Conference on Music Information Retrieval (ISMIR)*, 2005, pp. 666–671.
[46] M. I. Mandel and D. P. W. Ellis, "Song-level features and support vector machines for music classification," in *6th International Conference on Music Information Retrieval (ISMIR)*, 2005.
[47] D. Turnbull, L. Barrington, D. Torres, and G. Lanckriet, "Semantic annotation and retrieval of music and sound effects," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 2, pp. 467–476, 2008.
[48] J. Wang, X. Anguera, X. Chen, and D. Yang, "Enriching music mood annotation by semantic association reasoning," in *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1445–1450.
[49] E. Coviello, Y. Vaizman, A. B. Chan, and G. Lanckriet, "Multivariate autoregressive mixture models for music auto-tagging," in *13th International Conference on Music Information Retrieval (ISMIR)*, 2012, pp. 547–552.
[50] J.-J. Aucouturier and M. Sandler, "Segmentation of musical signals using hidden markov models," *Preprints-Audio Engineering Society*, 2001.
[51] A. Sheh and D. P. Ellis, "Chord segmentation and recognition using em-trained hidden markov models," in *4th International Conference on Music Information Retrieval (ISMIR)*, 2003, pp. 185–191.
[52] X. Shao, C. Xu, and M. S. Kankanhalli, "Unsupervised classification of music genre using hidden markov model," in *Multimedia and Expo, 2004. ICME 04. 2004 IEEE International Conference on*, vol. 3. IEEE, 2004, pp. 2023–2026.
[53] J. Reed and C.-H. Lee, "A study on music genre classification based on universal acoustic models." in *ISMIR*, 2006, pp. 89–94.
[54] G. Peeters, "Musical key estimation of audio signal based on hidden markov modeling of chroma vectors," in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, 2006, pp. 127–131.
[55] M. Hoffman, P. Cook, and D. Blei, "Data-driven recomposition using the hierarchical dirichlet process hidden markov model," in *Proc. International Computer Music Conference*. Citeseer, 2008.
[56] L. Barrington, A. B. Chan, and G. Lanckriet, "Modeling Music as a Dynamic Texture," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 3, pp. 602–612, Mar. 2010.
[57] Y. Vaizman, R. Y. Granot, and G. Lanckriet, "Modeling dynamic patterns for emotional content in music," in *12th International Conference on Music Information Retrieval (ISMIR)*, 2011, pp. 747–752.
[58] M. Barthet, G. Fazekas, and M. Sandler, "Multidisciplinary perspectives on music emotion recognition: Implications for content and context-based models," in *9th International Symposium on Computer Music Modeling and Retrieval (CMMR) Music and Emotions*, June 2012, pp. 19–22.
[59] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "A survey of audio-based music classification and annotation," *Multimedia, IEEE Transactions on*, vol. 13, no. 2, pp. 303–319, 2011.
[60] V. Imbrasaite, T. Baltrusaitis, and P. Robinson, "Emotion tracking in music using continuous conditional random fields and relative feature representation," in *ICME AAM Workshop*, 2013.
[61] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006. [Online]. Available: http://research.microsoft.com/en-us/um/people/cmbishop/books.htm
[62] J.-J. Aucouturier and F. Pachet, "Music similarity measures: Whats the use?" in *3rd International Conference on Music Information Retrieval (ISMIR)*, 2002, pp. 157–163.
[63] J. H. Jensen, D. P. W. Ellis, M. G. Christensen, and S. H. Jensen, "Evaluation of distance measures between gaussian mixture models of mfccs," in *8th International Conference on Music Information Retrieval (ISMIR)*, 2007.
[64] T. Jebara and A. Howard, "Probability Product Kernels," *Journal of Machine Learning Research*, vol. 5, pp. 819–844, 2004.
[65] R. D. Bock and J. V. Jones, *The measurement and prediction of judgment and choice*. Holden-day, 1968.
[66] K. Train, *Discrete Choice Methods with Simulation*. Cambridge University Press, 2009.
[67] J. Madsen, B. S. Jensen, J. Larsen, and J. B. Nielsen, "Towards predicting expressed emotion in music from pairwise comparisons." in *9th Sound and Music Computing Conference (SMC) Illusions*, July 2012.
[68] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized representer theorem," *Computational Learning Theory*, vol. 2111, pp. 416–426, 2001.
[69] J. Zhu and T. Hastie, "Kernel logistic regression and the import vector machine," in *Journal of Computational and Graphical Statistics*. MIT Press, 2001, pp. 1081–1088.
[70] F. Huszar, "A GP classification approach to preference learning," in *NIPS Workshop on Choice Models and Preference Learning*, 2011, pp. 1–4.
[71] J. Madsen, B. S. Jensen, and J. Larsen, "Predictive modeling of expressed emotions in music using pairwise comparisons," in *From Sounds to Music and Emotions*, ser. Lecture Notes in Computer Science, M. Aramaki, M. Barthet, R. Kronland-Martinet, and S. Ystad, Eds. Springer Berlin Heidelberg, 2013, vol. 7900, pp. 253–277.
[72] Y.-H. Yang and H. Chen, "Ranking-Based Emotion Recognition for Music Organization and Retrieval," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 762–774, May 2011.
[73] M. Müller and S. Ewert, "Chroma Toolbox: MATLAB implementations for extracting variants of chroma-based audio features," in *12th International Conference on Music Information Retrieval (ISMIR)*, Miami, USA, 2011.
[74] B.Mathieu, S.Essid, T.Fillon, J.Prado, and G.Richard, "an easy to use and efficient audio feature extraction software," in *11th International Conference on Music Information Retrieval (ISMIR)*, 2010.
[75] D. Sculley, "Web-scale k-means clustering," *International World Wide Web Conference*, pp. 1177–1178, 2010.

Jens Madsen Biography text here.

Bjørn Sand Jensen Biography text here.

Jan Larsen Biography text here.