Methods for Combining Statistical Models of Music

Marcus Pearce, Darrell Conklin, and Geraint Wiggins

Centre for Computational Creativity, City University,
Northampton Square, London EC1V 0HB, UK
{m.t.pearce,conklin,geraint}@city.ac.uk

Abstract. The paper concerns the use of multiple viewpoint representation schemes for prediction with statistical models of monophonic music. We present an experimental comparison of the performance of two techniques for combining predictions within the multiple viewpoint framework. The results demonstrate that a new technique based on a weighted geometric mean outperforms existing techniques. This finding is discussed in terms of previous research in machine learning.

1 Introduction

Statistical models of symbolically represented music have been used in a number of theoretical and practical applications in the computer modelling and retrieval of music. Examples of such applications include computer-assisted composition [1–3], machine improvisation with human performers [4, 5], music information retrieval [6], stylistic analysis of music [7–9] and cognitive modelling of music perception [10, 11]. A significant challenge faced in much of this research arises from the need to simultaneously represent and process many different features or attributes of the musical surface. One approach to this problem is to represent music within a framework that allows a musical object to be observed from *multiple viewpoints* [12, 13]. Multiple viewpoint modelling strategies take advantage of such a representational framework by deriving individual expert models for any given representational viewpoint and then combining the results obtained from each model. Here we consider multiple viewpoint systems from the perspective of statistical modelling and prediction of monophonic music. In particular, we are concerned with the evaluation of different methods for combining the predictions of different models in a multiple viewpoint system. To this end, we compare the performance of a previously reported combination technique based on a weighted arithmetic mean [14] with a new technique based on a weighted geometric mean.

Multiple viewpoint systems are a specific instance of a more general class of strategies in machine learning collectively known as *ensemble learning methods*. As noted in [15], ensemble methods can improve the performance of machine learning algorithms for three fundamental reasons. The first is statistical: with small amounts of training data it is often hard to obtain reliable performance

measures for a single model. By combining a number of well performing models, we can reduce the risk of inadvertently selecting models whose performance does not generalise well to new examples. The second reason is computational: for learning algorithms which employ local search, combining models which search locally from different starting points in the hypothesis space can yield better performance than any of the individual models. The final reason is representational: a combination of learning models may allow the system to reach parts of the hypothesis space that the individual models would be unable, or extremely unlikely, to reach. The development of multiple viewpoint systems was motivated largely by representational concerns arising specifically in the context of computer modelling of music [13]. Although ensemble methods have typically been applied in classification problems, as opposed to the prediction problems studied here, we shall draw on that body of work as required.

The paper is structured as follows. In §2, we review the theory of multiple viewpoints as a representational formalism, describe how we may develop statistical models within the multiple viewpoint framework and present the entropy based performance metrics that we shall use to assess the performance of our models. In §3, we introduce the techniques for combining viewpoint predictions and the experimental procedure that we use to evaluate them is described in §4. The results of our experiments are presented and discussed in §5. Finally, in §6, we conclude by suggesting some directions for future research.

2 Background

2.1 Representing Music with Multiple Viewpoints

In this section, we review the representation language of the multiple viewpoint framework as developed in [13, 14]. The specific motivation in the development of the framework was to extend the application of statistical modelling techniques to domains, such as music, where events have an internal structure and are richly representable in languages other than the basic event language. Here we consider the framework only insofar as it applies to monophonic music. See [16] for extensions to accommodate the representation of homophonic and polyphonic music.

The framework takes as its *musical surface* [17] sequences of musical events which roughly correspond to individual notes as notated in a score. Each event consists of a finite set of descriptive variables or *basic attributes* each of which may assume a value drawn from some finite domain or alphabet. Each attribute describes an abstract property of events and is associated with a type, τ, which specifies the properties of that attribute (see Table 1). Each type is associated with a syntactic domain, $[\tau]$, denoting the set of all syntactically valid elements of that type. Each type is also supplied with an informal semantics by means of an associated semantic domain, $[\![\tau]\!]$, which denotes the set of possible meanings for elements of type τ and a function, $[\![.]\!]_\tau : [\tau] \rightarrow [\![\tau]\!]$, which returns the semantic interpretation of any element of type τ. The Cartesian product of the domains of n basic types $\tau_1, \ldots, \tau_n$ is referred to as the *event space*, ξ:

Table 1. Sets and functions associated with typed attributes

Symbol	Interpretation	Example
τ	A typed attribute	`cpitch`
$[\tau]$	Syntactic domain of τ	$\{60,\ldots,72\}$
$\langle\tau\rangle$	Type set of τ	{`cpitch`}
$[\![\tau]\!]$	Semantic domain of τ	$\{C_4,C\sharp_4,\ldots,B_4,C_5\}$
$[\![.]\!]_\tau : [\tau] \rightarrow [\![\tau]\!]$	Semantic interpretation of $[\tau]$	$[\![60]\!]_{\texttt{cpitch}} = C_4$
$\Psi_\tau : \xi^* \rightharpoonup [\tau]$	see text	see text

$$\xi = [\tau_1] \times [\tau_2] \times \ldots \times [\tau_n]$$

An *event* $e \in \xi$ is an instantiation of the attributes $\tau_1,\ldots,\tau_n$ and consists of an n-tuple in the event space. The event space ξ, therefore, denotes the set of all representable events and its cardinality, $|\xi|$, will be infinite if one or more of the attribute domains $[\tau_1],\ldots,[\tau_n]$ is infinite. We shall use the notation $e_i^j \in \xi^*$ to denote a sequence of events $e_i,\ldots,e_j$ where $j \geq i \in \mathbb{Z}^+$ and ξ^* denotes the set of all sequences composed of members of ξ including the empty sequence ε.

A *viewpoint* modelling a type τ is a partial function, $\Psi_\tau : \xi^* \rightharpoonup [\tau]$, which maps sequences of events onto elements of type τ.[1] Each viewpoint is associated with a *type set* $\langle\tau\rangle \subseteq \{\tau_1,\ldots,\tau_n\}$, stating which basic types the viewpoint is derived from and is, therefore, capable of predicting [14]. A collection of viewpoints forms a *multiple viewpoint system*. We now describe the nature of several distinct *classes* of viewpoint which may be defined.

Basic Viewpoints For *basic types*, those associated with basic attribute domains, Ψ_τ is simply a projection function [14] and $\langle\tau\rangle$ is a singleton set containing just the basic type itself. An example of a basic type is one which represents the chromatic pitch of an event in terms of MIDI note numbers (`cpitch`; see Table 1).

Derived Viewpoints A type that does not feature in the event space but which is derived from one or more basic types is called a *derived type*. The function Ψ_τ acts as a *selector* function for events, returning the appropriate attribute value when supplied with an event sequence [14]. Since the function is partial the result may be undefined (denoted by $\perp$) for a given event sequence. Many of the derived types implemented in [14] are inspired by the construction of quotient GISs in [18]. The motivation for constructing such types is to capture and model the rich variety of relational and descriptive terms in a musical language [14]. A viewpoint modelling a derived type is called a *derived* viewpoint and the

[1] While viewpoints were defined in [13] to additionally comprise a statistical model of sequences in $[\tau]^*$, here we consider viewpoints to be a purely representational formalism and maintain a clear distinction between our representation language and our modelling strategies.

types from which it is derived, and which it is capable of predicting, are given by the type set for that viewpoint. An example of a derived viewpoint is one which represents melodic intervals in the chromatic pitch domain. Given the basic type `cpitch` shown in Table 1, the derived viewpoint `cpint` [14] is defined by the function:

$$\Psi_{\texttt{cpint}}(e_1^j) = \begin{cases} \perp & \text{if } j = 1, \\ \Psi_{\texttt{cpitch}}(e_j) - \Psi_{\texttt{cpitch}}(e_{j-1}) & \text{otherwise.} \end{cases}$$

Linked Viewpoints A system of viewpoints modelling primitive types will have limited representational and predictive power due to its inability to represent any interactions between those individual types [13]. *Linked viewpoints* are an attempt to address this problem and were motivated by the direct product GISs described in [18]. A *product type* $\tau = \tau_1 \otimes \ldots \otimes \tau_n$ between n constituent types $\tau_1, \ldots, \tau_n$ has the following properties:

$$\begin{aligned} [\tau] &= [\tau_1] \times \ldots \times [\tau_n] \\ \langle \tau \rangle &= \bigcup_{k=1}^{n} \langle \tau_k \rangle \\ [\![\tau]\!] &= [\![\tau_1]\!] \textit{ and } \ldots \textit{ and } [\![\tau_n]\!] \\ \Psi_\tau(e_1^j) &= \begin{cases} \perp & \text{if } \Psi_{\tau_i}(e_1^j) \text{ is undefined for any } i \in \{1, \ldots, n\} \\ \langle \Psi_{\tau_1}(e_1^j), \ldots, \Psi_{\tau_n}(e_1^j) \rangle & \text{otherwise.} \end{cases} \end{aligned}$$

A linked viewpoint is one which models a product type. Linked viewpoints add to the representation language the ability to represent disjunctions of conjunctions of attribute values (as opposed to simple disjunctions of attribute values). To give an example, it was found in [13] that a viewpoint linking melodic pitch interval with inter-onset interval (`cpint` $\otimes$ `ioi`) proved useful in modelling the chorale melodies harmonised by J. S. Bach. This finding suggests that these two attributes types are correlated in that corpus.

Test Viewpoints A *test viewpoint* models a Boolean-valued attribute type and is used to define locations in a sequence of events [19]. An example is the `fib` viewpoint defined in [14] as follows:

$$\Psi_{\texttt{fib}}(e_1^j) = \begin{cases} \text{T} & \text{if } \Psi_{\texttt{posinbar}}(e_1^j) = 1, \\ \text{F} & \text{otherwise} \end{cases}$$

where `posinbar` is a derived type giving the relative position of an event in the bar (e.g., $[\![1]\!]_{\texttt{posinbar}}$ = the first event in the current bar).

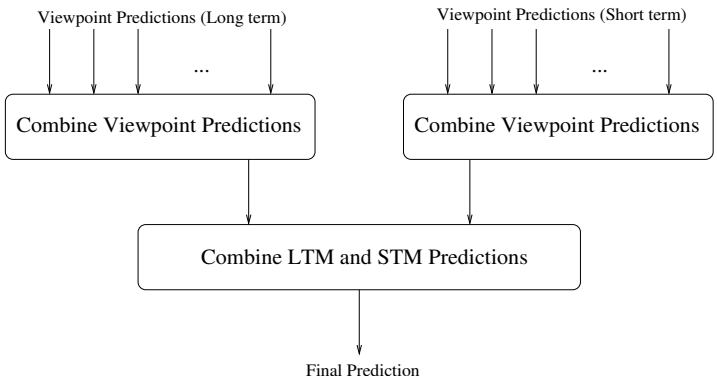


Fig. 1. The architecture of a multiple viewpoint system

Threaded Viewpoints Types whose values are only defined at certain points in a piece of music (e.g., the first event in each bar) are called *threaded types* and viewpoints modelling these types are called *threaded viewpoints*. Threaded viewpoints model the value of a *base viewpoint* at temporal or metric locations where a specified test viewpoint returns true and are undefined otherwise [19]. The base viewpoint may be any primitive or linked viewpoint. Threaded viewpoints were developed to take advantage of structure emerging from metrical grouping and phrasing in music. The alphabet of a threaded viewpoint is the Cartesian product of the alphabets of the base viewpoint and a viewpoint, `ioi`, representing inter-onset intervals [19]. To take an example, consider the `thrbar` viewpoint defined in [13] constructed from the base viewpoint `cpint` and the test viewpoint `fib`. This viewpoint represents the melodic intervals between the first events in each consecutive bar and is undefined at all other locations in a melodic sequence. Its viewpoint elements consist of pairs of `cpint` and `ioi` elements corresponding to the pitch interval between the first events in two successive bars and the inter-onset interval between those events.

2.2 Modelling Music with Multiple Viewpoints

The Overall Architecture For our purposes, a statistical model associated with a viewpoint τ is a function m_τ which accepts a sequence of events in τ^* and which returns a distribution over $[\tau]$ reflecting the estimated conditional probabilities of the identity of the next viewpoint element in the sequence (see [13, 20] for further description of the nature of such models). Examples of such models include n-gram models which have been used for automatic classification of musical works [9], polyphonic score retrieval [6] and modelling of music perception [10, 11], and dictionary based statistical models which have been used for automatic music classification [7], computer improvisation with human performers [4] and computer-assisted composition [2].

A predictive system operating on a multiple viewpoint representation language consists of a number of models $m_{\tau_1}, \ldots, m_{\tau_n}$ corresponding to the collec-

tion of viewpoints $\tau_1, \ldots, \tau_n$ in the multiple viewpoint system. For each viewpoint, we actually employ two models: a *long-term model* (LTM) and a *short-term model* (STM). The LTM is trained on the entire training corpus while the STM is constructed online for each composition modelled and is discarded after the relevant composition has been processed. The motivation for using an STM is to take advantage of recently occurring sequences whose structure and statistics may be specific to the individual composition being predicted. The use of an STM has been found to improve the prediction performance of multiple viewpoint models of music [14, 20]. The predictions of both long- and short-term models must be combined to produce a final prediction (see §3). A number of general architectures can be envisaged to achieve this combination:

1. combine the STM and LTM predictions for each viewpoint individually and then combine the resulting viewpoint predictions;
2. combine the viewpoint predictions separately for the long- and short-term models and then combine the resulting LTM and STM predictions;
3. combine all long- and short-term viewpoint predictions in a single step.

We follow previous research [13] in choosing the second of these alternatives (see Figure 1). Two additional issues arise from the fact that our models accept sequences in $[\tau]^*$ rather than ξ^* and return distributions over $[\tau]$ rather than ξ: first, the corpus of event sequences in ξ^* must be preprocessed into sequences in τ^* which are used to train the models; and second, the resulting distribution over $[\tau]$ must be postprocessed into a distribution over ξ so it may be combined with distributions generated by other models. These issues are discussed in turn.

Preprocessing the Event Sequences We may convert sequences in ξ^* to sequences in $[\tau]^*$ using the function $\Phi_\tau : \xi^* \rightarrow [\tau]^*$ [13] such that:

$$\Phi_\tau(e_1^i) = \begin{cases} \varepsilon & \text{if } e_1^i = \varepsilon \\ \Phi_\tau(e_1^{i-1}) & \text{if } \Psi_\tau(e_1^i) = \perp \\ \Phi_\tau(e_1^{i-1})\Psi_\tau(e_i) & \text{otherwise} \end{cases}$$

Since $\Psi_\tau(e_1^i) = \perp \Rightarrow \Phi_\tau(e_1^i) = \Phi_\tau(e_1^{i-1})$, it is necessary to check that $\Psi_\tau(e_1^i)$ is defined to prevent the same sequence in $[\tau]^*$ being added to the model more than once [13].

Completion of a Multiple Viewpoint System A model m_τ returns a distribution over $[\tau]$ but, in order to combine the distributions generated by the models for different viewpoints, we need to convert them into distributions over the basic event space ξ. In the interests of efficiency, prediction is elicited in stages, one for each basic type of interest [14]. Only those viewpoints which contain in their type set the basic type, τ_b, currently under consideration are activated at each stage. The conversion is achieved by a function which maps elements of $[\tau]$ onto elements of $[\tau_b]$:

$$\Psi'_\tau : \xi^* \times [\tau] \rightarrow P([\tau_b])$$

where $P(S)$ denotes the power set of set S. The function Ψ'_τ is implemented by creating a set of events each of which corresponds to a distinct basic element in $[\tau_b]$. A set of sequences is created by appending each of these events to the sequence of previously processed events in the composition. By calling the function Ψ_τ on each of these sequences each element in $[\tau_b]$ is put into the mapping with the current element of $[\tau]$. The mapping is, in general, many-to-one since a derived sequence $\Phi_\tau(e_1^i)$ could represent many sequences of events other than e_1^i. As a result, the probability estimate returned by the model for the derived sequence must be divided equally among the basic event sequences onto which it maps.

A model m_τ must return a complete distribution over the basic attributes in $\langle\tau\rangle$. This does not present problems for basic viewpoints where the viewpoint domain is predefined to be the set of viewpoint elements occurring in the corpus.[2] However, for derived viewpoints, such as `cpint`, it may not be possible to derive a complete distribution over [`cpitch`] from the set of derived elements occurring in the corpus. To address this problem, the domain of each derived type τ is set prior to prediction of each event such that there is a one-to-one correspondence between $[\tau]$ and the domain of the basic type $\tau_b \in \langle\tau\rangle$ currently being predicted. We assume that the modelling technique has some facility for assigning probabilities to events that have never occurred before [13, 20]. If no viewpoints predict some basic attribute then the completion of that attribute must be predicted on the basis of information from other sources or on the basis of a uniform distribution over the attribute domain. In this research, m_{τ_b} was used to achieve the completion of attribute τ_b in such cases.

Once the distributions generated by each model in a multiple viewpoint system have been converted to complete distributions over the domain of a basic type, the distributions may be combined into final distributions for each basic type. The topic of this paper is how best to achieve this combination and two methods are discussed in detail in §3.

2.3 Performance Metrics

Given a probability mass function $p(a \in \mathcal{A}) = P(\mathcal{X} = a)$ of a random variable $\mathcal{X}$ distributed over a discrete alphabet $\mathcal{A}$, the entropy is calculated as:

$$H(p) = H(\mathcal{X}) = -\sum_{a \in \mathcal{A}} p(a) \log_2 p(a). \tag{1}$$

Shannon's (1948) fundamental coding theorem states that entropy provides a lower bound on the average number of binary bits per symbol required to encode

[2] The domain of a viewpoint modelling the onset time of events is potentially infinite and assumes a value derived from the onset time of the previous event and the set of inter-onset intervals that occur in the corpus [13].

an outcome of the variable $\mathcal{X}$. The corresponding upper bound occurs in the case where each symbol in the alphabet has an equal probability of occurring, $\forall a \in \mathcal{A}, p(a) = \frac{1}{|\mathcal{A}|}$, as shown in Equation 2.

$$H_{max}(p) = H_{max}(\mathcal{A}) = \log_2 |\mathcal{A}| \tag{2}$$

Entropy has an alternative interpretation in terms of the degree of uncertainty that is involved in selecting a symbol from an alphabet: greater entropy implies greater uncertainty. In practise, we rarely know the true probability distribution of the stochastic process and use a model to approximate the probabilities in Equation 1. *Cross entropy* is a quantity which represents the divergence between the entropy calculated from these estimated probabilities and the source entropy. Given a model which assigns a probability of $p_m(a_1^j)$ to a sequence a_1^j of outcomes of $\mathcal{X}$, we can calculate the cross entropy $H(p_m, a_1^j)$ of model m with respect to event sequence a_1^j as shown in Equation 3.

$$H(p_m, a_1^j) = -\frac{1}{j} \sum_{i=1}^{j} \log_2 p_m(a_i | a_1^{i-1}) \tag{3}$$

While cross entropy provides a direct measure of performance in the field of data compression, it has a wider use in the evaluation of statistical models. Since it provides us with a measure of how uncertain a model is, on average, when predicting a given sequence of events, it can be used to compare the performance of different models on some corpus of data [21, 22].

3 Combining Viewpoint Prediction Probabilities

3.1 Introduction

In this section, we shall describe several techniques for combining the distributions generated by statistical models for different viewpoints. Let τ_b be the basic viewpoint currently under consideration and $[\tau_b] = \{t_1, t_2, \ldots, t_k\}$ its domain. Our multiple viewpoint system has n viewpoints $\tau_1, \ldots, \tau_n$ which are derived from τ_b and there exist corresponding sets of long-term models $LTM = \{ltm_1, ltm_2, \ldots, ltm_n\}$ and short-term models $STM = \{stm_1, stm_2, \ldots, stm_n\}$. We require a function that combines the distributions over τ_b generated by sets of models. As described in §2.2, this function is used in the first stage of combination to combine the distributions generated by the LTM and the STM separately and, in the second stage of prediction, to combine the two combined distributions resulting from the first stage. In what follows we describe functions for combining individual probabilities which may then be applied to sorted distributions over τ_b. For the purposes of illustration, we employ an anonymous set of models $M = m_1, m_2, \ldots, m_n$.[3]

[3] We refer to combination schemes based on the arithmetic mean as arithmetic combination and those based on the geometric mean as geometric combination. Similar

3.2 Arithmetic Combination

Perhaps the simplest method of combining distributions is to compute the arithmetic mean of the estimated probabilities for each symbol $t \in [\tau_b]$ such that:

$$p(t) = \frac{1}{n} \sum_{m \in M} p_m(t).$$

This combination technique may be improved by weighting the contributions made by each of the models such that:

$$p(t) = \frac{\sum_{m \in M} w_m p_m(t)}{\sum_{m \in M} w_m}.$$

A method for calculating the weights, w_m, is described in [14]. It is based on the entropies of the distributions generated by the individual models such that greater entropy (and hence uncertainty) is associated with a lower weight. The weight of model m is $w_m = H_{relative}(p_m)^{-b}$. The *relative entropy* $H_{relative}(p_m)$ of a model is given by:

$$H_{relative}(p_m) = \begin{cases} H(p_m)/H_{max}(p_m) & \text{if } H_{max}([\tau_b]) > 0 \\ 1 & \text{otherwise.} \end{cases}$$

where H and H_{max} are as defined in Equations 1 and 2 respectively. The bias $b \in \mathbb{Z}$ is a parameter giving an exponential bias towards models with lower relative entropy. Note that with $b = 0$, the weighted arithmetic scheme is equivalent to its non-weighted counterpart. This weighting mechanism is described in more detail in [14] where the weighted arithmetic mean was used for combining both viewpoint predictions and the predictions of the long- and short-term models while this method was used for combining viewpoint predictions only in [13].[4]

3.3 Geometric Combination

We present a novel method for combining the distributions generated by our statistical models which is based on a weighted geometric mean. A simple geometric mean of the estimated probabilities for each symbol $t \in [\tau_b]$ is calculated as:

$$p(t) = \frac{1}{R} \prod_{m \in M} p_m(t)^{\frac{1}{n}}.$$

distinctions have been made in the literature between linear and logarithmic opinion pools [23], combining classifiers by averaging and multiplying [24] and mixtures and products of experts [25, 26].

[4] Other methods were also examined in [13] including a ranking-based combination method as well as a method based on the rule of combination used in the Dempster-Shafer theory of evidence.

Table 2. The basic and derived viewpoints used in this research

τ	Class	$[\![.]\!]_\tau$	$[\tau]$	$\langle\tau\rangle$
`onset`	basic	onset time of event	{0,1,2,...}	{`onset`}
`cpitch`	basic	chromatic pitch (MIDI)	{60,...,79,81}	{`cpitch`}
`ioi`	derived	inter-onset interval	{1,...,20}	{`onset`}
`fib`	derived	(not) first event in bar	{T,F}	{`onset`}
`cpint`	derived	sequential melodic interval	$\mathbb{Z}$	{`cpitch`}
`cpintfref`	derived	vertical interval from referent	{0,...,11}	{`cpitch`}

where R is a normalisation constant. As in the case of the arithmetic mean, this technique may be improved by weighting the contributions made by each of the models such that:

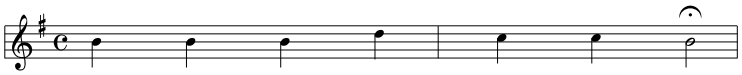
$$p(t) = \frac{1}{R} \prod_{m \in M} p_m(t)^{w_m}$$

where R is a normalisation constant and the weights w_m are normalised such that they sum to one. We may use the same weighting technique as for arithmetic combination (see §3.2) and, once again, with $b = 0$, the weighted geometric scheme is equivalent to its non-weighted counterpart.

4 Experimental Procedure

The corpus of music used is a subset of the chorale melodies harmonised by J. S. Bach. A set of 185 chorales (BWV 253 to BWV 438) has been encoded by Steven Rasmussen and is freely available in the `**kern` format [27] from the *Centre for Computer Assisted Research in the Humanities* (CCARH) at Stanford University, California (see http://www.ccarh.org). We have used cross entropy, as defined in Equation 3, computed by 10-fold cross-validation [28, 29] over the corpus as a performance metric for our models. The statistical model used was a smoothed, variable-order n-gram model described in more detail in [20]. Since the goal of this research was to examine methods for combining viewpoint predictions, we have used a constant set of viewpoints corresponding to the best performing of the multiple viewpoint systems described in [13]. This system consists of the following viewpoints:

$$\begin{aligned}
&\texttt{cpintfref} \otimes \texttt{cpint},\\
&\texttt{cpint} \otimes \texttt{ioi},\\
&\texttt{cpitch},\\
&\texttt{cpintfref} \otimes \texttt{fib}.
\end{aligned}$$



`onset`	0	24	48	72	96	120	144
`cpitch`	71	71	71	74	72	72	71
`ioi`	⊥	24	24	24	24	24	24
`fib`	T	F	F	F	T	F	F
`cpint`	⊥	0	0	3	-2	0	-1
`cpintfref`	4	4	4	7	5	5	4
`cpint`⊗`ioi`	⊥	(0 24)	(0 24)	(3 24)	(-2 24)	(0 24)	(-1 24)

Fig. 2. The first phrase of the chorale melody *Meinen Jesum laß' ich nicht, Jesus* (BWV 379) represented as viewpoint sequences in terms of the basic, derived and linked viewpoints used in the experiments

It is capable of modelling the basic type `cpitch` alone. See Table 2 for details of each of the viewpoints in this system and Figure 4 for an exemplary use of these viewpoints in representing an excerpt from a chorale melody in terms of viewpoint sequences. We have examined the weighted arithmetic and geometric combination schemes described in §3 in both stages of combination with the bias settings drawn from the set {0,1,2,3,4,5,6,7,8,16,32}.[5]

5 Results and Discussion

The results of the experiment are shown in Table 3 which is divided into four sections corresponding to the four combinations of the two combination methods. Figures in bold type represent the lowest entropies in each of the four sections of the table. The results are also plotted graphically in Figure 5. The first point to note is that the multiple viewpoint system is capable of predicting the dataset with much lower entropies (e.g., 2.045 bits/symbol) than those reported in [20] for a system modelling chromatic pitch alone (e.g., 2.342 bits/symbol) on the same corpus. This replicates the findings of [13] and lends support to the assertion that the multiple viewpoint framework can increase the predictive power of statistical models of music. It is also clear that the use of an entropy based weighting scheme improves performance and that performance can be further improved by tuning the bias parameter which gives exponential bias towards models with lower relative entropies [14].

[5] The Dempster-Shafer and rank-based combination schemes described in [13] were found to perform less well than these two methods (when optimally weighted) and are not included in the results.

Table 3. Cross entropies (bits/symbol) of the data given the model using weighted arithmetic and geometric schemes with a range of bias settings for combining the LTM-STM and viewpoint predictions

		Viewpoint Combination																				
		Arithmetic											Geometric									
		0	1	2	3	4	5	6	7	8	16	32	0	1	2	3	4	5	6	7	8	16	32
A	0	2.493	2.437	2.393	2.363	2.342	2.327	2.316	2.309	2.304	2.290	2.291	2.357	2.321	2.299	2.286	2.278	2.274	2.271	2.270	2.269	2.274	2.285
r	1	2.434	2.368	2.317	2.281	2.257	2.241	2.230	2.222	2.217	2.207	2.212	2.256	2.216	2.192	2.180	2.175	2.173	2.173	2.174	2.175	2.188	2.203
i	2	2.386	2.317	2.264	2.229	2.207	2.193	2.184	2.178	2.175	2.171	2.178	2.189	2.150	2.130	2.123	2.122	2.124	2.126	2.130	2.133	2.152	2.169
t	3	2.350	2.279	2.228	2.196	2.177	2.166	2.160	2.156	2.155	2.159	2.168	2.146	2.111	2.096	2.094	2.097	2.102	2.107	2.112	2.117	2.142	2.160
h	4	2.323	2.253	2.204	2.175	2.159	2.150	2.147	2.145	2.146	2.157	2.169	2.119	2.088	2.077	2.078	2.085	2.092	2.100	2.107	2.113	2.142	2.161
m	5	2.303	2.234	2.188	2.161	2.147	2.141	2.139	2.140	2.141	2.158	2.173	2.102	2.074	2.066	2.070	2.079	2.089	2.098	2.106	2.113	2.146	2.167
e	6	2.288	2.221	2.176	2.152	2.140	2.136	2.135	2.137	2.140	2.161	2.179	2.091	2.066	2.060	2.066	2.077	2.088	2.098	2.108	2.116	2.151	2.174
t	7	2.276	2.211	2.168	2.146	2.136	2.133	2.134	2.136	2.140	2.165	2.184	2.085	2.061	2.057	2.064	2.076	2.088	2.099	2.110	2.118	2.156	2.180
i	8	2.268	2.204	2.163	2.142	2.133	**2.131**	2.133	2.136	2.140	2.168	2.189	2.080	2.057	2.055	2.064	2.076	2.089	2.101	2.112	2.121	2.161	2.186
c	16	2.243	2.186	2.152	2.136	2.132	2.133	2.138	2.143	2.149	2.184	2.212	2.073	**2.053**	2.054	2.066	2.081	2.097	2.111	2.123	2.134	2.182	2.212
	32	2.239	2.185	2.154	2.140	2.138	2.140	2.145	2.151	2.157	2.195	2.226	2.074	2.055	2.058	2.070	2.086	2.103	2.118	2.132	2.143	2.194	2.226
	0	2.496	2.437	2.386	2.346	2.316	2.294	2.278	2.266	2.257	2.237	2.240	2.311	2.267	2.238	2.222	2.213	2.208	2.207	2.206	2.207	2.219	2.234
G	1	2.425	2.354	2.295	2.252	2.222	2.202	2.188	2.178	2.172	2.160	2.165	2.200	2.155	2.129	2.118	2.114	2.114	2.116	2.119	2.122	2.141	2.157
e	2	2.372	2.298	2.240	2.201	2.176	2.161	2.151	2.145	2.142	2.142	2.150	2.138	2.098	2.081	2.077	2.079	2.084	2.090	2.096	2.101	2.126	2.143
o	3	2.334	2.260	2.206	2.172	2.152	2.141	2.135	2.133	2.132	2.141	2.154	2.104	2.070	2.059	2.060	2.067	2.076	2.084	2.092	2.099	2.13	2.149
m	4	2.307	2.235	2.185	2.155	2.139	2.131	2.128	2.128	2.129	2.146	2.163	2.086	2.057	2.050	2.054	2.064	2.075	2.085	2.095	2.103	2.138	2.159
e	5	2.288	2.218	2.171	2.145	2.132	2.127	2.126	2.127	2.130	2.152	2.171	2.077	2.051	2.046	2.053	2.065	2.077	2.089	2.099	2.108	2.146	2.169
t	6	2.275	2.207	2.163	2.139	2.129	**2.125**	2.126	2.128	2.132	2.158	2.179	2.072	2.048	**2.045**	2.054	2.067	2.080	2.092	2.103	2.113	2.154	2.178
r	7	2.265	2.200	2.158	2.136	2.127	**2.125**	2.127	2.130	2.134	2.163	2.186	2.069	2.047	**2.045**	2.055	2.069	2.083	2.096	2.107	2.117	2.160	2.185
i	8	2.258	2.194	2.155	2.134	2.127	**2.125**	2.128	2.132	2.136	2.167	2.192	2.068	2.047	2.046	2.057	2.071	2.085	2.099	2.111	2.121	2.165	2.191
c	16	2.240	2.184	2.151	2.136	2.132	2.133	2.138	2.144	2.150	2.186	2.216	2.070	2.051	2.053	2.065	2.081	2.098	2.112	2.125	2.136	2.186	2.217
	32	2.239	2.185	2.154	2.141	2.138	2.141	2.146	2.151	2.158	2.198	2.229	2.073	2.055	2.057	2.070	2.087	2.104	2.12	2.134	2.145	2.197	2.230

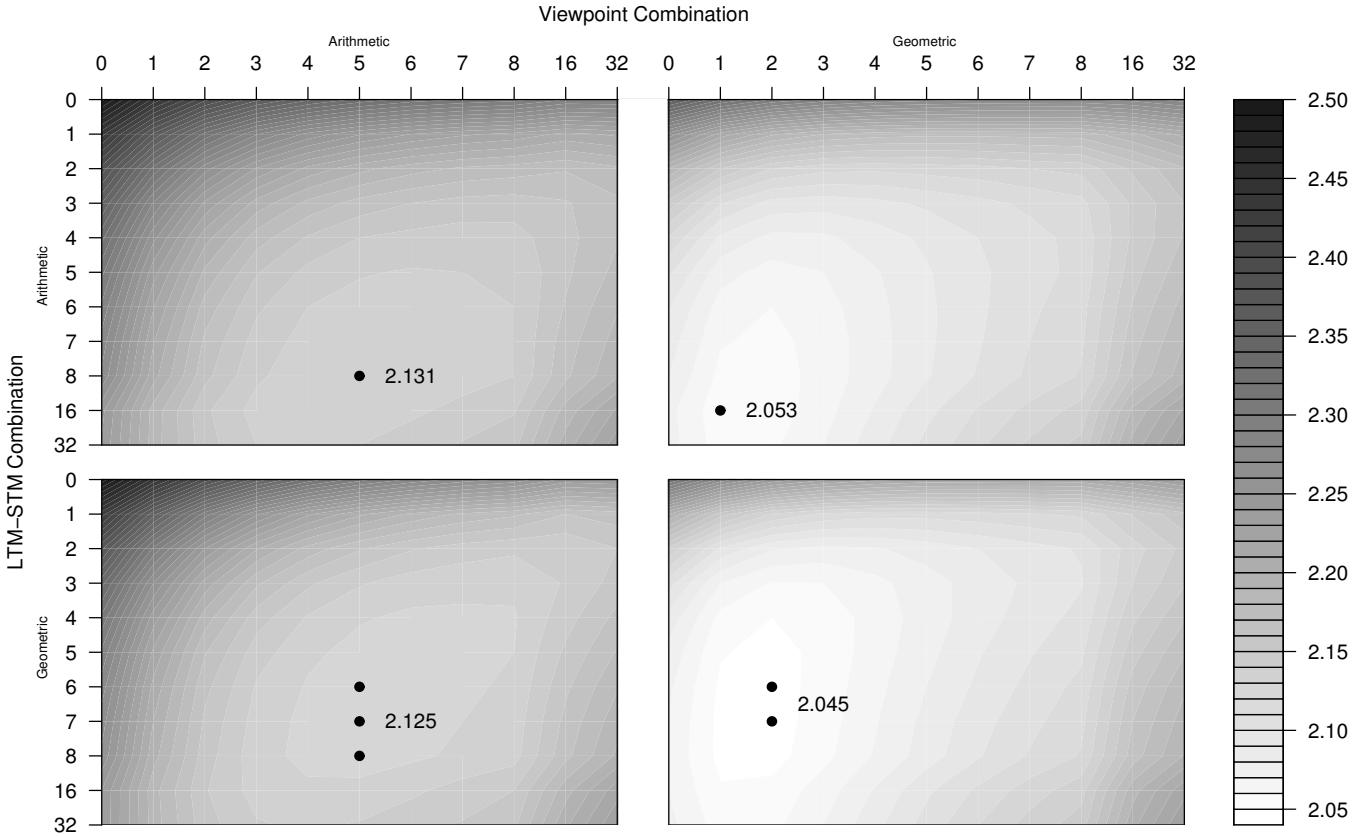


Fig. 3. Cross entropies (bits/symbol) of the data given the model using weighted arithmetic and geometric schemes with a range of bias settings for combining the LTM-STM and viewpoint predictions

Regarding the combination methods, the results demonstrate that the weighted geometric combination introduced in this paper tends to outperform arithmetic combination and that this effect is much more marked in the case of viewpoint combination than it is for LTM-STM combination. Some theoretical justification for this result can be found in the literature on combining classifier systems. Hinton [25, 26] argues that combining distributions through multiplication has the attractive property of making distributions "sharper" than the component distributions. For a given element of the distributions it suffices for just one model to correctly assign that element a low estimated probability. If this is the case, the combined distribution will assign that element a low probability regardless of whether other models (incorrectly) assign that element a high estimated probability. Arithmetic combination, on the other hand, will tend to produce combined distributions that are less sharp than the component distributions and is prone to erroneously assigning relatively high estimated probabilities to irrelevant elements. However, since the combined distribution cannot be sharper than any of the component distributions arithmetic combination has the desirable effect of suppressing estimation errors [24].

In [24] the performance of arithmetic and geometric combination schemes is examined in the context of multiple classifier systems. In accordance with theoretical predictions, an arithmetic scheme performs better when the classifiers operate on identical data representations and a geometric scheme performs better when the classifiers employ independent data representations. Analogously, we hypothesise that when combining viewpoint predictions (derived from distinct data representations), a geometric scheme performs better since it trusts specialised viewpoints to correctly assign low probability estimates to a given element. Consider movement to a non scale degree as an example: a model associated with `cpitch` might return a high probability estimate for such a transition whereas a model associated with `cpintfref` is likely to return a low estimated probability. In cases such as this, it is preferable to trust the model operating over the more specialised data representation (i.e., `cpintfref`).

When combining LTM-STM predictions (where each distribution is already the result of combining the viewpoint predictions), on the other hand, a premium is placed on minimising estimation errors. For example, for n-grams which are common in the current composition but rare in the corpus as a whole, the LTM will return low estimates and the STM high estimates. In cases such as this, it is preferable to suppress the estimation errors yielded by the LTM. The finding that geometric combination still outperforms arithmetic combination in LTM-STM combination may be a result of the fact that n-grams are added online to the LTM as prediction progresses much as they are for the STM [20]. Finally, it is possible that the difference in relative performance of the geometric and arithmetic schemes for LTM-STM and viewpoint combination is a result of the order in which these combinations are performed (see Figure 1). However, we hypothesise that this is not the case and the observed pattern of results arises from the difference between combining distributions derived from distinct data representations as opposed to combining two distributions already combined

from the same sets of representations. Further research is required to examine these hypotheses in more depth.

Another aspect of the results that warrants discussion is the effect on performance of the bias parameter which gives an exponential bias towards distributions with lower relative entropy. Overall performance seems to be optimised when the bias for LTM-STM combination is relatively high (between 6 and 16) and the bias for viewpoint combination is relatively low (between 1 and 5). We suggest that this is due to the fact that at the beginning of a composition, the STM will generate relatively high entropy distributions due to the lack of context. In this case, it will be advantageous for the system to strongly bias the combination towards the LTM predictions. This is not an issue when combining viewpoint predictions and more moderate bias values tend to be optimal. Other research has also found that high bias values for the combination of the LTM-STM predictions tend to improve performance leading to the suggestion that the weight assigned to the STM could be progressively increased from an initially low value at the beginning of a composition as more events are processed [14].

The results shown in Table 2 also reveal an inverse relationship between the optimal bias settings for LTM-STM combination and those for viewpoint combination. With high bias values for LTM-STM combination, low bias values for viewpoint combination tend to be optimal and vice versa. High bias settings will make the system bolder in its estimation by strongly favouring sharper distributions while low bias settings will lead it to more conservative predictions. On these grounds, with all other things being equal, we would expect moderate bias values to yield optimal performance. If an extreme bias setting is preferred in one stage of combination for some other reason (e.g., the case of LTM-STM combination just discussed), the negative effects may, it seems, be counteracted to some extent by using settings at the opposing extreme in the other stage. Although these arguments are general, we would expect the optimal bias settings themselves to vary with different data, viewpoints and predictive systems.

6 Conclusions

We have presented an experimental comparison of the performance of two techniques for combining distributions within the multiple viewpoint framework for representing and modelling music. Specifically, a novel combination technique based on a weighted geometric mean was compared to an existing technique based on a weighted arithmetic mean. We have used an entropy based technique to compute the weights which accepts a parameter which fine-tunes the exponential bias given to distributions with lower relative entropy. A range of parameterisations of the two techniques have been evaluated using cross entropy computed by 10-fold cross-validation over a dataset of chorale melodies harmonised by J. S. Bach. The results demonstrate that the weighted geometric combination introduced in this research tends to outperform arithmetic combination especially for the combination of viewpoint models. Drawing on related findings in previous research in machine learning on combining multiple classi-

fiers, it was hypothesised that this asymmetry arises from the difference between combining distributions derived from distinct data representations as opposed to combining distributions derived from the same data representations.

We would like to conclude the paper by suggesting some directions we feel would be fruitful for future research. Perhaps the most important limitation of this research is that results have been obtained for a single dataset (representing a single genre of melodic music) using a single set of viewpoints. However, this research does make specific hypotheses to be refuted or corroborated by further experiments which go beyond these restrictions. Our confidence in the generality of these results obtained would be increased if they could be replicated using different corpora of music, different viewpoint systems and other forms of music. It would also be useful to conduct a thorough examination of the effect of the overall architecture of the system on performance. How is performance affected, for example, if we first combine the LTM-STM predictions for each viewpoint and then combine the resulting distributions? It seems unlikely that a single combination of all distributions will improve performance but this conjecture can only be tested by empirical experimentation. Finally, it remains to be seen whether other combination schemes developed in the field of machine learning [30–32] can be profitably applied to modelling music with multiple viewpoint systems.

This research has examined a number of techniques for improving the prediction performance of statistical models of music. These techniques have been evaluated in an application neutral manner using cross entropy as an index of model uncertainty. In statistical language modelling, it has been demonstrated that cross entropy provides a good predictor of model performance in specific practical contexts:

> "For a number of natural language processing tasks, such as speech recognition, machine translation, handwriting recognition, stenotype transcription and spelling correction, language models for which the cross entropy is lower lead directly to better performance."
>
> [21, p. 39].

While corresponding results are not currently available in the literature on computational music research, we believe the techniques presented in this paper can be applied profitably to practical problems in the modelling and retrieval of music.

References

1. Ames, C.: The Markov process as a compositional model: a survey and tutorial. Leonardo **22** (1989) 175–187
2. Assayag, G., Dubnov, S., Delerue, O.: Guessing the composer's mind: applying universal prediction to musical style. In: Proceedings of the 1999 International Computer Music Conference, San Francisco: ICMA (1999) 496–499

3. Hall, M., Smith, L.: A computer model of blues music and its evaluation. Journal of the Acoustical Society of America **100** (1996) 1163–1167
4. Lartillot, O., Dubnov, S., Assayag, G., Bejerano, G.: Automatic modelling of musical style. In: Proceedings of the 2001 International Computer Music Conference, San Francisco: ICMA (2001) 447–454
5. Rowe, R.J.: Machine composing and listening with Cypher. Computer Music Journal **16** (1992) 43–63
6. Pickens, J., Bello, J.P., Monti, G., Sandler, M.B., Crawford, T., Dovey, M., Byrd, D.: Polyphonic score retrieval using polyphonic audio queries: A harmonic modeling approach. Journal of New Music Research **32** (2003) 223–236
7. Dubnov, S., Assayag, G., El-Yaniv, R.: Universal classification applied to musical sequences. In: Proceedings of the 1998 International Computer Music Conference, San Francisco: ICMA (1998) 332–340
8. Ponsford, D., Wiggins, G.A., Mellish, C.: Statistical learning of harmonic movement. Journal of New Music Research **28** (1999) 150–177
9. Westhead, M.D., Smaill, A.: Automatic characterisation of musical style. In Smith, M., Smaill, A., Wiggins, G., eds.: Music Education: an Artificial Intelligence Approach, Berlin, Springer (1993) 157–170
10. Ferrand, M., Nelson, P., Wiggins, G.: A probabilistic model for melody segmentation. In: Electronic Proceedings of the 2nd International Conference on Music and Artificial Intelligence (ICMAI'2002), University of Edinburgh, Scotland (2002)
11. Eerola, T.: The Dynamics of Musical Expectancy: Cross-cultural and Statistical Approaches to Melodic Expectations. PhD thesis, Faculty of Humanities, University of Jyväskylä, Finland (2004) Jyväskylä Studies in Humanities, 9.
12. Ebcioğlu, K.: An expert system for harmonising four–part chorales. Computer Music Journal **12** (1988) 43–51
13. Conklin, D., Witten, I.H.: Multiple viewpoint systems for music prediction. Journal of New Music Research **24** (1995) 51–73
14. Conklin, D.: Prediction and entropy of music. Master's thesis, Department of Computer Science, University of Calgary (1990) Available as Technical Report 1990–390–14.
15. Dietterich, T.G.: Ensemble methods in machine learning. In: First International Workshop on Multiple Classifier Systems. Lecture Notes in Computer Science, New York: Springer Verlag (2000) 1–15
16. Conklin, D.: Representation and discovery of vertical patterns in music. In Anagnostopoulou, C., Ferrand, M., Smaill, A., eds.: Proceedings of the Second International Conference of Music and Artificial Intelligence. (2002) 32–42
17. Jackendoff, R.: Consciousness and the Computational Mind. MIT Press, Cambridge, MA (1987)
18. Lewin, D.: Generalised Musical Intervals and Transformations. Yale University Press, New Haven/London (1987)
19. Conklin, D., Anagnostopoulou, C.: Representation and discovery of multiple viewpoint patterns. In: Proceedings of the 2001 International Computer Music Conference, San Francisco: ICMA (2001)
20. Pearce, M.T., Wiggins, G.A.: Improved methods for statistical modelling of monophonic music. To appear in *Journal of New Music Research* (2004)
21. Brown, P.F., Della Pietra, S.A., Della Pietra, V.J., Lai, J.C., Mercer, R.L.: An estimate of an upper bound on the entropy of English. Computational Linguistics **18** (1992) 32–40
22. Manning, C.D., Schütze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge, MA (1999)

23. Genest, C., Zidek, J.V.: Combining probability distributions: a critique and an annotated bibliography. Statistical Science **1** (1986) 114–148
24. Tax, D.M.J., van Breukelen, M., Duin, R.P.W., Kittler, J.: Combining multiple classifiers by averaging or by multiplying? Pattern Recognition **33** (2000) 1475–1485
25. Hinton, G.: Products of experts. In: Proceedings of the Ninth International Conference on Artificial Neural Networks. Volume 1. (1999) 1–6
26. Hinton, G.: Training products of experts by minimizing contrastive divergence. Technical Report GCNU TR 2000-004, Gatsby Computational Neuroscience Unit, UCL (2000)
27. Huron, D.: *Humdrum* and *Kern*: selective feature encoding. In Selfridge-Field, E., ed.: Beyond MIDI: The Handbook of Musical Codes. MIT Press, Cambridge, MA (1997) 375–401
28. Mitchell, T.M.: Machine Learning. McGraw Hill, New York (1997)
29. Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation **10** (1998) 1895–1924
30. Xu, L., Krzyzak, A., Suen, C.Y.: Methods of combining multiple classifiers and thier applications to handwriting recognition. IEEE Transactions on Systems, Man and Cybernetics **22** (1992) 418–435
31. Chen, K., Wang, L., Chi, H.: Methods of combining multiple classifiers with different features and their applications to text-independent speaker identification. International Journal of Pattern Recognition and Artificial Intelligence **11** (1997) 417–415
32. Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On combining classifiers. IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998) 226–239