

# Melodic analysis with segment classes

Darrell Conklin

Received: 28 September 2005 / Revised: 17 February 2006 / Accepted: 20 March 2006 / Published online:  
21 June 2006  
Springer Science + Business Media, LLC 2006

**Abstract** This paper presents a representation for melodic segment classes and applies it to music data mining. Melody is modeled as a sequence of segments, each segment being a sequence of notes. These segments are assigned to classes through a knowledge representation scheme which allows the flexible construction of abstract views of the music surface. The representation is applied to sequential pattern discovery and to the statistical modeling of musical style.

**Keywords** Music analysis · Music data mining · Statistical models · Music representation · Segmentation

## 1. Introduction

Music analysis is concerned with describing the structures in a piece of music and discovering how they relate to one another and change through time. The process of music analysis can be viewed at a high level as data mining; the classification of structures, and the discovery of sequential predictive rules that relate context to continuation.

Music has many special properties that make it an interesting and challenging problem for data mining research. It is temporal, highly structured, and polyphonic music has concurrent streams of events which together form harmonic structures. Events in music exist at higher levels than just notes, for example as phrases or chords. Events have time spans and are not just points, as assumed by time series analysis and sequential pattern mining methods. Like natural language, music has a deep structure with extensive dependencies between events not adjacent at the musical surface. Music has repetition within individual pieces, and musical styles have recurrent structures at more abstract levels between different pieces. Similar musical material arises through transformation of the musical surface or by alternative realization of abstract music structures.

---

**Editor:** Gerhard Widmer

---

D. Conklin (✉)  
Department of Computing, City University, London, United Kingdom  
e-mail: conklin@city.ac.uk

Data mining can also contribute much to music analysis. Data mining can be applied to an individual piece, to a corpus of pieces, or to a combination of both. The attention of music analysis has been mainly to the level of a particular piece rather than to general style (Brown & Dempster, 1989). Comparative analysis, the growing field of music analysis concerned with corpus analysis (Cook, 1987), can be productively explored with data mining approaches (Huron, 1996). These studies have so far been concerned mainly with classical descriptive statistics and there is clearly scope for the contribution of modern data mining including classification and pattern discovery methods.

Essential to any data mining application is a primary decision on the features used to describe objects. In music, a useful knowledge representation called *viewpoints* has been used to represent abstract properties of musical notes for statistical modeling of melody (Conklin and Witten, 1995). The representation provides syntactic constructors used to build new features from more primitive ones. Recently Conklin and Anagnostopoulou (2006) described how the viewpoints representation can be extended to describe pieces of music at higher levels in terms of sequences of abstract classes assigned to musical segments.

The formation of groups from a musical stream is a fundamental part of the music perception process (Lerdahl & Jackendoff, 1983), and for music analysis the segmentation of the musical surface is a prelude to revealing organizational principles of a piece. It is therefore important to be able to describe abstract properties of segments and how the segments relate to one another. This description is the concern of the general method of *musical set theory* (Forte, 1973), where different collections of notes, equivalent under transposition or inversion, are related by having the same interval vector representation. Understanding relations between segments is also the basis of *semiotic analysis* (Nattiez, 1975; Cook, 1987), where the temporal flow of segment classes (*paradigmatic types*) is viewed in a sequential (*syntagmatic*) dimension. Computer implementations of various aspects of semiotic analysis—mainly for the unsupervised clustering of melodic segments—have been proposed (Anagnostopoulou & Westermann, 1997; Cambouropoulos & Widmer, 2000; Pienimäki & Lemström, 2004).

The objective of this paper is to demonstrate how the viewpoints representation for segment classes can be applied to interesting music data mining tasks. Melody is modeled as a sequence of segments, each segment itself a sequence of notes. The segmentation of music comes with a price for data mining; the sparse data problem becomes more serious if notes are grouped. It is necessary to represent segments in a way both abstract enough to reveal frequent patterns, but not so abstract as to reveal nothing of statistical significance in the data. In this paper, some suitably abstract viewpoints for segment classes are defined and applied.

Two data mining applications of segment classes are presented in this paper. It is shown how interesting sequential association rules, relating contexts of segment classes to continuation, can be discovered. Following this, after a detailed description of statistical modeling using segment classes, it is shown how the representation can be used for music style discrimination in a Bayesian classification framework.

## 2. Music representation

For computational music analysis, it is important to be able to structure and represent the musical surface as voices, segments, and simultaneities. For this purpose, an algebraic data type can be used to represent music in a hierarchical fashion (Hudak et al., 1996; Balaban, 1996; Marsden, 2000; Conklin, 2002). A *music object* is a note (type **Note**), or (recursively) a sequence (type **Seq**) of music objects (other structures such as simultaneities are possible but

**Table 1** Glossary of notation

<b>Note</b>	a note music object
<b>Seq(X)</b>	regular sequences containing events of type $X$
<b>Seq(Note)</b>	sequences of notes: called <i>segments</i>
<b>Seq(Seq(Note))</b>	sequences of segments
$[\ ]$	the empty sequence
$\overline{x_n}$	a sequence $[x_1, \dots, x_n]$ , with $\overline{x_0} = [\ ]$
$\tau$	a viewpoint
$\tau(\overline{w_n})$	the class of event $w_n$ in the context $\overline{w_{n-1}}$
$\widehat{\tau}(\overline{w_n})$	the viewpoint sequence for event sequence $\overline{w_n}$
$\perp$	the undefined event class

not needed in this paper). All music objects have a *duration*. When music objects are added to a sequence, they acquire an *onset time* relative to the beginning of the sequence; they have time spans and are called *events*. Sequences of events are ordered by increasing onset time. Music objects may have other basic attributes besides duration; for example, notes have a *pitch* usually represented by MIDI number, but might have further detail in terms of pitch spelling (e.g., the spellings G#4 and Ab4 both have MIDI number 68). The additional basic attributes used will depend on the particular analysis task.

Music object representation allows a score to be structured in many useful ways, but this work is concerned only with scores structured as *regular* sequences, where all events within the sequence are of the same type. The parametric type **Seq(X)** denotes regular sequences containing events exclusively of the type given by parameter  $X$ . For example, a sequence of notes has type **Seq(Note)**, and a segmented melody has type **Seq(Seq(Note))**. A sequence  $[x_1, \dots, x_n]$  of  $n$  items will be abbreviated as  $\overline{x_n}$  (see Table 1: Glossary of notation).

In this paper, melodies will be segmented either by the natural grouping of notes at annotated phrase boundaries, or by taking into account musical metre. In the latter approach, segments are created at every metrical grid line, with notes split and duplicated if their time spans are intersected by a metrical grid line. The metrical grids used in this paper will be at the quarter note beat level and at the bar level.

### 3. Knowledge representation of event classes

The technique of *viewpoints* was developed to address the need for multiple simultaneous representations for music, at varying levels of abstraction. For data mining, they provide a flexible technique for handling the sparse data problem, because events are mapped into more abstract classes with the effect of yielding more reliable counts from a corpus (Conklin and Witten, 1995).

In terms of representing classes of events and relations between events, the viewpoints method is an example of a *vivid* knowledge representation scheme (Brachman & Levesque, 2004) in that event features and relations are computed directly from the underlying music object data structure (Section 2) using functions that compute event abstractions. A viewpoint  $\tau$  is simply a function that computes, for a regular sequence  $\overline{w_n}$  of type **Seq(X)**, a value representing the class of the last event  $w_n$  of  $\overline{w_n}$  given its sequential context  $\overline{w_{n-1}}$ . The domain of  $\tau$  is therefore **Seq(X)**, and a *melodic viewpoint* is one with domain **Seq(Note)**, while a *segmental viewpoint* is one with domain **Seq(Seq(Note))**.

A viewpoint partitions the space of events with the same context into equivalence classes represented by elements of its range set. Not all events in a sequence need to have a defined

class for a particular viewpoint (for example, the first note in a melody has no preceding note and therefore no melodic interval class); all such events are mapped to a special undefined class which is denoted by the symbol  $\perp$ .

Much of the utility of the representation depends on the transformation of sequences of concrete events to sequences of abstract event classes; the transformed sequence is called a *viewpoint sequence*. For a viewpoint  $\tau$  and an event sequence  $\overline{w}_n$  the viewpoint sequence, denoted  $\widehat{\tau}(\overline{w}_n)$ , is the transformed sequence comprising the mapping of  $\tau$  to each of the prefixes of  $\overline{w}_n$ :

$$[\tau(\overline{w}_1), \dots, \tau(\overline{w}_n)]$$

with the constraint that no uninformative  $\perp$  classes are present in the result. The notations  $\tau$  and  $\widehat{\tau}$  correspond to the  $\Psi_\tau$  and  $\Phi_\tau$  notations used by Conklin and Witten (1995).

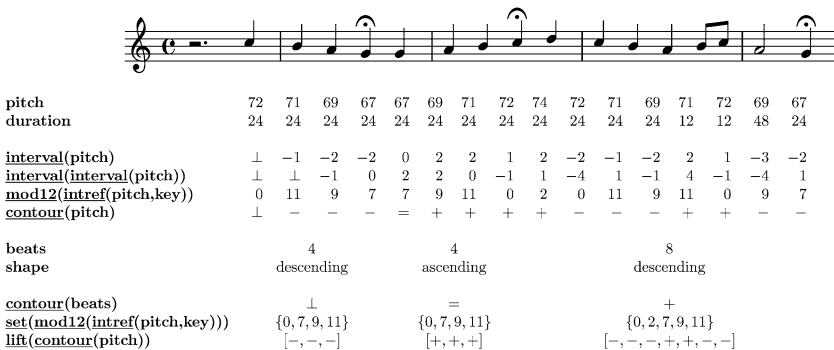
The representation would not be so useful if new functions had to be designed and implemented for every new conception of properties and relations of events in music. Rather, a set of *constructors* with a simple syntax is used to facilitate defining new features in a declarative fashion (Conklin and Witten, 1995). Conklin and Anagnostopoulou (2006) further extended the viewpoints representation by developing some new constructors for building segmental viewpoints; two used in this paper are:

- lift**( $\tau$ ) computes the viewpoint sequence for a segment using viewpoint  $\tau$ . In this way the viewpoint sequence itself is treated as the class of a segment;
- set**( $\tau$ ) computes the *set* of classes in the range of  $\tau$  occurring in a segment (in contrast to **lift**( $\tau$ ), which computes the *sequence* of classes in the range of  $\tau$ ).

The remainder of this section gives two illustrations of the viewpoints representation method for music.

### 3.1. Example 1

Figure 1 shows an illustration of the viewpoints method for the first three phrases of a Bach chorale melody in C major (BWV 255). At the top are two basic melodic viewpoints **pitch** (MIDI number) and **duration** (24 ticks per quarter note). This is followed by some composite melodic viewpoints, using constructors such as **interval** (the difference between classes for an



**Fig. 1** Examples of viewpoints for the first three phrases of a Bach chorale melody (BWV 255). Viewpoint constructors are underlined

**Table 2** Segment shape classes, relating the initial and final pitches  $a$  and  $c$  of a segment to the mean  $b$  of its internal pitches, or relating  $a$  and  $c$  for segments with only two notes

Relation	Class	Relation	Class
$a < b < c$	ascending	$a < c$	ascending
$a > b > c$	descending	$a > c$	descending
$a = b = c$	horizontal	$a = c$	horizontal
$a < b = c$	ascending-horizontal	$a > b = c$	descending-horizontal
$a < b > c$	convex	$a > b < c$	concave
$a = b < c$	horizontal-ascending	$a = b > c$	horizontal-descending

event and its immediately preceding event); **intref** (the difference between the two indicated classes, here the pitch and the key of the piece); **mod12** (modulo 12 transformation); **contour** (sign function on an integer). Following this is a primitive viewpoint **beats** measuring the number of quarter note beats between the downbeat of the segment and the end of the last bar of the segment, and a primitive viewpoint **shape**. For this, the nine phrase shape classes originally defined by Huron (1996) are expanded here to include three additional rules for segments with only two notes (see Table 2). Finally in Fig. 1 are three composite segmental viewpoints, two using the **set** and **lift** constructors which construct segmental viewpoints from melodic viewpoints. The second construction computes the set of intervals from a reference pitch (in this case, the key of the piece) in a modulo 12 system. The last computes the melodic contour viewpoint sequence of a segment, and is another abstract way to represent the overall shape of a phrase.

In summary, viewpoints can be basic (selecting basic event attributes), primitive (computed from basic attributes), or composite (built using viewpoint constructors). They can be melodic (applying to notes) or segmental (applying to segments). Though some melodic viewpoints in Fig. 1 represent familiar musical features—melodic interval, melodic contour, and pitch classes—most are novel constructions made possible by the viewpoint constructor method.

### 3.2. Example 2

As a second illustration of music representation using viewpoints, Fig. 2 shows a segmental viewpoint sequence for the right-hand part of the Bach two-part invention in A minor (BWV 784). The piece was segmented using a metrical grid at every quarter-note beat and transformed to a lifted melodic contour viewpoint sequence (recall Fig. 1); three bars (12 segments) of the 25 bars in the piece are shown in Fig. 2. This viewpoint representation can be compared to the paradigmatic analysis provided by Höthker, Hörnel and Anagnostopoulou (2001). In that study, the 100 segments in the piece were classified by a human music analyst using predominately melodic contour features. The lifted contour viewpoint sequence has a close agreement (Höthker distance of 0.02 over the 100 segments) with the human analysis.



**Fig. 2** The first three bars of Bach two-part invention BWV 784 (right-hand part) showing the lifted contour sequence for a metrical grid segmentation at every quarter-note beat

**Table 3** Melodic datasets used, with counts of various structures within the corpus. Beats refers to the number of quarter note beats in the corpus

Description	Pieces	Phrases	Bars	Beats	Notes
Nova Scotia folk songs	152	885	2535	7911	8551
Bach Chorale melodies	185	1141	2745	10347	9227

This example illustrates that meaningful paradigmatic types for music segments can emerge naturally from suitably abstract viewpoint representations.

#### 4. Music data mining with segment classes

This section provides two examples of the use of viewpoints to model aspects of melodic segments in music corpora. The first shows how segment classes can be used as a basis for inter-opus sequential association rule discovery. The second example presents the topic of statistical modeling of music and shows how event sequence models can be naturally extended, using segmental viewpoints, to include the modeling of segments and not only notes as events.

Two music corpora are used in this section (Table 3): the Nova Scotia folk songs (Creighton, 1966) and the 185 Bach chorale melodies. Data is available from Center for Computer Assisted Research in the Humanities ([www.ccarh.org](http://www.ccarh.org)). For all examples, pieces are first converted to viewpoint sequences for a selected viewpoint and a suffix tree is built from these transformed sequences. The suffix tree data structure allows for rapid recovery of repeated and recurrent patterns in the data.

An annotated segmentation for phrases is available for both corpora. For the Nova Scotia corpus, phrase boundaries were manually added by comparing the score data with the lyrics (Creighton, 1966) and placing a phrase boundary at the end of every line of lyrics. In addition, various other segmentations were explored, as will be discussed in this section. Four melodies in the corpora have internal time signature changes that were correctly handled for metrical grid segmentation at the bar level.

##### 4.1. Sequential association rules

The discovery of recurrent or repeating patterns in music has been addressed by many researchers (Conklin & Anagnostopoulou, 2001; Rolland & Ganascia, 2000; Hsu, Liu & Chen, 2001; Cambouropoulos, 1998; Cope, 1991; Lartillot, 2004; Meredith, Lemström & Wiggins, 2002). Conklin and Anagnostopoulou (2006) presented the idea of segmental pattern discovery, where the fundamental structures of analysis are not notes but segments. This section presents a new application of segmental pattern discovery, where segmental patterns are used as a basis for expressing sequential association rules.

A *pattern* is a viewpoint sequence that recurs in a corpus. A pattern *occurs* in a piece if it is a substring of the viewpoint sequence of the piece. In a corpus, the *piece count* of a pattern is the number of pieces in which the pattern occurs at least once. The *total count* of a pattern is its total number of occurrences in a corpus, including overlapping repetitions within a piece. A *sequential association rule* derived from a pattern  $\overline{c_k}$  is an implication  $\overline{c_{k-1}} \Rightarrow c_k$ , meaning that the class  $c_k$  tends to follow pattern  $\overline{c_{k-1}}$ . The *confidence* of the rule is  $p(c_k | \overline{c_{k-1}})$ ; the

**Table 4** Sequential association rules discovered in the Nova Scotia folk songs

Rule	Antecedent	Consequent	tc:pc	Conf	Lift
1.1	[ascending]	descending	63:48	0.34	1.4
1.2	[descending, ascending]	descending	18:14	0.43	1.7
1.3	[ascending, descending, ascending]	descending	10:10	0.62	2.5
2.1	[concave]	descending	43:31	0.34	1.4
2.2	[descending, concave]	descending	16:13	0.52	2.1
2.3	[concave, descending, concave]	descending	7:6	0.78	3.2
3	[ascending, convex, ascending]	descending	5:5	0.45	1.8
4	[convex, convex, ascending]	descending	8:8	0.4	1.6
5	[ascending-horizontal]	convex	5:5	0.71	1.9
6	[convex, concave]	convex	17:12	0.5	1.3
7	[ascending, ascending]	ascending	9:6	0.31	1.5
8.1	[concave, descending]	concave	9:8	0.21	1.5
8.2	[descending, concave, descending]	concave	5:5	0.31	2.2
9	[ascending, concave]	concave	6:6	0.27	1.9
10	[concave, convex]	concave	8:5	0.23	1.6

Legend: tc, pc: total/piece count; conf: rule confidence.

total count of  $\overline{c_k}$  divided by the total count of  $\overline{c_{k-1}}$ . The *lift* of the rule is defined as

$$\frac{p(c_k | \overline{c_{k-1}})}{p(c_k)}.$$

The rule lift measures the degree of correlation between the antecedent and the consequent of the rule. Association rules with positive correlation (lift greater than 1) can be efficiently found by traversing a suffix tree representation of viewpoint sequences from a corpus.

#### 4.1.1. Results

To illustrate inter-opus association rule discovery, the Nova Scotia folk songs were segmented on phrase boundaries, and pieces were transformed into **shape** viewpoint sequences (Table 2). Association rules were derived from patterns with a piece count of at least 5, and those with a lift at least 1.25 were retained. Table 4 illustrates the rules discovered in the Nova corpus, with instances of rule specialization indicated.

Rules 1.1 and 2.1 correspond precisely to two significant rules discovered by Huron (1996) by querying a large database of folk songs for pairs of phrases with the second having a descending shape. The rules with a concave, ascending, or horizontal-ascending antecedent are claimed to reveal a melodic arch which spans two phrases and peaks at the end of the first phrase. The results presented here reveal the presence of two of these rules in the smaller Nova Scotia corpus, and also point out more specific versions. For example, rule 2.3 represents a periodic repetition of rule 2.1, extended to four successive phrases. A number of other rules are found, including the frequent rule 6 which describes a convex shape following convex and concave antecedent phrases, and rule 7 which describes a succession of three ascending phrases. All rules are significant implications that can be used for the prediction of phrase shape from context.

### 4.2. Statistical models

Statistical models of event sequences have been used for both classification and generation of music. For music classification, models can represent and be trained on styles, genres, composers, or generally any set of related pieces. For music generation, they can be used for either style imitation (Conklin, 2003) or in real-time music improvisation systems (Assayag & Dubnov, 2004; Pachet, 2003).

Statistical models of music fall roughly into two categories based on their data representation; models of whole-piece features (Ponce de León & Iñesta, 2003) and models of event sequences (Conklin and Witten, 1995; Conklin, 2003; Pearce, 2005). Whole-piece models describe the piece using variables such as number of notes, note density, and pitch range. Event models, on the other hand, compute the probability of a piece as a product of the conditional probabilities of individual events in the piece. This section shows how event models can be constructed for sequences of melodic segments using viewpoints. The fundamentals of statistical models of word sequences (the rough analogy of “word” in language to “event” in music should be held in mind) in natural language are now reviewed, then viewpoint models will be presented.

Statistical models of word sequences assign a probability

$$p(\overline{w}_n) = p([w_1, \dots, w_n]) = \prod_{i=1}^n p(w_i | \overline{w}_{i-1})$$

to a sequence  $\overline{w}_n$ , with the equivalence following from the chain rule of probability. Estimation of statistical models from sparse data requires that various conditional independence assumptions are made. A common method is to condition only on short suffixes of the context  $\overline{w}_{i-1}$  for each event  $w_i$ , leading to the family of *n-gram models*. A further refinement is based on the idea of clustering of words into equivalence classes. Rather than modeling the concrete sequence directly, words are mapped onto abstract classes by a function (say,  $\tau$ ) and the joint distribution with this abstract class sequence is modeled; the assumption being that abstract word classes will be far less sparse than concrete words in a corpus. Since words have only one class, the sequence probability  $p(\overline{w}_n)$  can be expressed as the product of the two terms  $p(\widehat{\tau}(\overline{w}_n))$  and  $p(\overline{w}_n | \widehat{\tau}(\overline{w}_n))$ , with further conditional independence assumptions leading to the *class-based n-gram model* (Brown et al., 1992):

$$p(\overline{w}_n) = \prod_{i=1}^n p(\tau(w_i) | \widehat{\tau}(\overline{w}_{i-1})) \times p(w_i | \tau(w_i)). \tag{1}$$

The *class probability*  $p(\tau(w_i) | \widehat{\tau}(\overline{w}_{i-1}))$  is handled with an n-gram model of class sequences, and the *class conditional probability*  $p(w_i | \tau(w_i))$  estimated using maximum likelihood from training data.

For music corpora, the sparse data problem of statistical modeling can be managed with viewpoints (Section 3), using a technique similar to the class-based *n-gram models* described above, though with a few important differences. First, in music the class of an event may be determined not only by the identity of the event but also by its sequential context. Therefore for some viewpoints it will be necessary to condition both terms of (1) on some contextual events. For example, a melodic interval class will have a zero probability if it pushes the pitch of the previous event beyond the range of the voice. Therefore, a melodic interval class probability needs to be conditioned on the previous event, and to determine the class



conditional probability of an event it is necessary to know the pitch of the previous event. The amount of context needed to determine a class will vary for different viewpoints. Putting this together yields the *viewpoint model*:

$$p(\overline{w}_n) = \prod_{i=1}^n p(\tau(\overline{w}_i) \mid \widehat{\tau}(\overline{w}_{i-1}), \overline{w}_{i-1}) \times p(w_i \mid \tau(\overline{w}_i), \overline{w}_{i-1}). \quad (2)$$

The formulation in (2) is general and applies regardless of the viewpoint and the associated type of regular sequence under consideration.

In viewpoint models for music, the class conditional distribution is not estimated from training data due to the additional dependencies in the model and due to the limited size of music corpora. Rather, uniform distributions are assumed, and the class conditional probability can be computed through the determination of the number of distinct events generated by the class  $\tau(\overline{w}_i)$  given the context  $\overline{w}_{i-1}$ . This computation is straightforward for melodic viewpoints and can be computed by simple enumeration, since the number of possible notes will be small and finite. For segmental viewpoints, the set is not explicitly enumerated because the number of events that can be generated by a particular segment class can be exponential in the length of the segment (for example, consider the number of distinct phrases of a certain length and shape, or having a certain pitch class set). The method used to determine the number of events generated by a segment class will vary for different primitive segmental viewpoints and constructors.

#### 4.2.1. Results

The evaluation of statistical models of word sequences is usually according to entropy or perplexity (Jurafsky & Martin, 2000). Here, the alternative idea of evaluating statistical event models by music style discrimination is explored. The 185 Bach chorale melodies and Nova Scotia folk songs were segmented in various ways, pieces were transformed into various viewpoint sequences, and viewpoint models were created for each style. For selected viewpoints, variable-length  $n$ -gram models with backoff probabilities and discounting (Jurafsky & Martin, 2000) were constructed for class probabilities. Discounting was used to allocate some probability space to both undefined ( $\perp$ ) classes and to unseen classes (those not encountered in the corpus). If either condition is signalled, the event is directly generated using a uniform distribution over basic attributes (by recursion down to the note level, if necessary). Pieces were classified using a Bayesian framework with uniform priors; the predicted style is from the model which generates the piece with highest probability according to (2). A full leave-one-out cross validation was applied; all traces of the test piece, including its contribution to the zero-order class vocabulary, were erased prior to predicting the style of the test piece.

The performance of the method on the style discrimination task, under various different melodic segmentations, is given in Table 5. At the bottom of the table, for comparison, is a simple melodic interval viewpoint which applies to unsegmented melodies. Most segmental viewpoints achieve accuracies significantly above the accuracy of 55% that could be attained by simply guessing the most probable style in the training set. This indicates that significant statistical regularities exist in the sequences of segment classes for each corpus. As expected, the shape viewpoint is not so effective when referring to the shape of a whole piece, but achieves greater accuracy when applied to smaller structures such as phrases

**Table 5** Accuracy of viewpoint models on style discrimination on the Bach chorale melodies and the Nova Scotia folk songs; segmental viewpoints (top), and the melodic interval viewpoint (bottom)

Viewpoint	Segmentation	Accuracy
<b>shape</b>	bar	65
	phrase	64
	piece	48
<b>set(mod12(intref(pitch,key)))</b>	note	92
	beats	98
	bar	91
	phrase	88
	piece	77
<b>lift(contour(pitch))</b>	beats	73
	bar	79
<b>interval(pitch)</b>	n/a	94

and bars. It is interesting that a segmental viewpoint based on pitch class sets achieves a slightly higher accuracy than a melodic viewpoint. Interestingly, it is at the level of a fine-grained segmentation—at the quarter note beat level—where this is achieved. The lack of improvement at coarser levels of segmentation, for example at the bar level, could be partly due to sparse data effects. The results suggest that the lifted contour viewpoint is a better discriminator of style than is the more abstract shape viewpoint.

## 5. Discussion and conclusion

Much work on data mining of melody has used sequences of notes as a representation. This paper proposed how music structured as sequences of segments can be used for data mining. A knowledge representation of segment classes was developed, and this was applied to two music data mining tasks. The approach uses music structured as regular sequences, where all events within the sequence are of the same type. Notes were grouped to form segments, though the representation and methods will naturally support any depth of regular segmentation hierarchy.

In natural language, statistical modeling has mainly been applied to modeling word sequences (Jurafsky & Martin, 2000). The idea of using class models for phrases rather than words was presented by Ries, Buø and Waibel (1996), with the result called *class phrase models*. There, words were grouped into short phrases, and the phrase class is the sequence of word classes in the phrase. The analogy to the work described here is revealed by considering words as notes and phrases as segments. The phrase representation used by Ries, Buø and Waibel (1996) can be viewed as an instance of the **lift** viewpoint constructor, where the basic event attributes rather than abstract event classes are lifted. Therefore the viewpoint representation may prove useful in natural language modeling.

In the statistical modeling results using segmental viewpoints, the lengths of component segments did not contribute to the overall probability of a segmented melody. For the corpora used here, this is reasonable since the lengths of various structures (phrases, bars, beats) are similar between the two corpora. For other style classification tasks, segment length may be more predictive of musical style. The use of phoneme segment length distributions has been given careful attention in the context of the family of *segment models* for speech recognition (Ostendorf, Digalakis & Kimball, 1996), where the class conditional distribution for segments is also conditioned on their length.

In the work of Bod (2002) on automated melody segmentation, a phrase class is created for each distinct phrase (modulo transposition) in a corpus. Phrase classes are soft (a phrase can be generated by more than one class), and a parsing algorithm is used to compute the segmentation and phrase class sequence with maximal probability. The class conditional probability of a phrase given a phrase class is also conditioned on the number of phrases in the piece. In a viewpoint modeling scheme, as described in this paper, hard classes are used and there are additional dependencies between adjacent phrase classes, as captured by the  $n$ -gram segment class model.

In formal music analysis, segmentation is the initial step from which subsequent description of structures in the piece and their relationships are made. This paper did not consider the general problem of automated segmentation, and assumed that the segmentation is provided or unambiguously computed. The statistical modeling scheme, however, could be used within a parsing algorithm for the segmentation process, taking segment lengths into account.

The representation of segments by the set of properties of component events has similarities to work on statistical models of harmony and polyphonic musical structures (Pickens & Crawford, 2002; Dubnov et al., 2003; Temperley, 2004). In these works, voicing or temporal information in a simultaneity is often discarded in favor of more abstract representations in terms of pitch class profiles. Constructors similar to the **set** constructor used in this paper could perform similar abstractions for polyphonic music.

The viewpoint modeling method introduced a general approach for handling event classes when the set of possible events generated by a segment class is too large to be enumerated. Uniform distributions were used, but seeing segment classes as non-uniform generators of sequences introduces the idea of viewpoints ranging over submodel classes, where the classes represent identifiers of statistical models (Galescu & Allen, 2000). Such a hierarchical modeling scheme would provide an alternative way to grouping similar segments into the same class. Future work will describe and develop this and the automated segmentation and polyphony ideas introduced above, in the context of real-time music improvisation systems.

## References

- Anagnostopoulou, C., & Westermann, G. (1997). Classification in music: A computational model for paradigmatic analysis. In *Proceedings of the International Computer Music Conference* (pp. 125–128). Thessaloniki.
- Assayag, G., & Dubnov, S. (2004). Using factor oracles for machine improvisation. *Soft Computing*, 8, 604–610.
- Balaban, M. (1996). The music structures approach in knowledge representation for music processing. *Computer Music Journal*, 20(2), 96–111.
- Bod, R. (2002). Memory-based models of melodic analysis: Challenging the Gestalt principles. *Journal of New Music Research*, 31(1), 27–37.
- Brachman, R., & Levesque, H. (2004). *Knowledge representation and reasoning*. Morgan Kaufmann.
- Brown, M., & Dempster, D. J. (1989). The scientific image of music theory. *Journal of Music Theory*, 33(1), 65–106.
- Brown, P., Della Pietra, V., deSouza, P., Lai, J., & Mercer, R. (1992). Class-based  $n$ -gram models of natural language. *Computational Linguistics*, 18(4), 467–479.
- Cambouropoulos, E. (1998). Towards a general computational theory of musical structure. PhD thesis, Faculty of Music, University of Edinburgh.
- Cambouropoulos, E., & Widmer, G. (2000). Motivic analysis via melodic clustering. *Journal of New Music Research*, 29(4), 347–370.
- Conklin, D. (2002). Representation and discovery of vertical patterns in music. In C. Anagnostopoulou, M. Ferrand, and A. Smaill (Eds.), *Music and artificial intelligence: Lecture notes in artificial intelligence 2445* (pp. 32–42). Springer-Verlag.

- Conklin, D. (2003). Music generation from statistical models. In *Proceedings of the AISB Symposium on Artificial Intelligence and Creativity in the Arts and Sciences* (pp. 30–35). Aberystwyth.
- Conklin, D., & Anagnostopoulou, C. (2001). Representation and discovery of multiple viewpoint patterns. In *Proceedings of the International Computer Music Conference* (pp. 479–485). Havana.
- Conklin, D., & Anagnostopoulou, C. (2006). Segmental pattern discovery in music. *INFORMS Journal on Computing*, 18(3).
- Conklin, D., & Witten, I. (1995). Multiple viewpoint systems for music prediction. *Journal of New Music Research*, 24(1), 51–73.
- Cook, N. (1987). *A guide to musical analysis*. Oxford University Press.
- Cope, D. (1991). *Computers and musical style*. A-R Editions, Madison, WI.
- Creighton, H. (1966). *Songs and ballads from nova scotia*. New York: Dover Publications, Inc.
- Dubnov, S., Assayag, G., Lartillot, O., & Bejerano, G. (2003). Using machine-learning methods for musical style modeling. *IEEE Computer*, 36(10), 73–80.
- Forte, A. (1973). *The structure of atonal music*. Yale University Press.
- Galescu, L., & Allen, J. (2000). Hierarchical statistical language models: Experiments on in-domain adaptation. In *Proceedings of the International Conference on Spoken Language Processing* (pp. 186–189). Beijing.
- Höthker, K., Hörnel, D., & Anagnostopoulou, C. (2001). Investigating the influence of representations and algorithms in music classification. *Computers and the Humanities*, 35, 65–79.
- Hsu, J.-L., Liu, C.-C., & Chen, A. (2001). Discovering nontrivial repeating patterns in music data. *IEEE Transactions on Multimedia*, 3, 311–325.
- Hudak, P., Makuvecich, T., Gadde, S., & Whong, B. (1996). Haskore music notation—An algebra of music. *Journal of Functional Programming*, 6(3), 465–483.
- Huron, D. (1996). The melodic arch in Western folksongs. *Computing in Musicology*, 10, 3–23.
- Jurafsky, D., & Martin, J. (2000). *Speech and language processing*. Englewood Cliffs, NJ: Prentice-Hall.
- Lartillot, O. (2004). An adaptive and multi-parametric approach for motivic pattern discovery. In *Proceedings of the Sound and Music Computing Conference* (pp. 117–124). Paris.
- Lerdahl, F., & Jackendoff, R. (1983). *A generative theory of tonal music*. Cambridge, MA: MIT Press.
- Marsden, A. (2000). *Representing musical time: A temporal logic approach*. Swets and Zeitlinger.
- Meredith, D., Lemström, K., & Wiggins, G. (2002). Algorithms for discovering repeated patterns in multidimensional representations of polyphonic music. *Journal of New Music Research*, 31(4), 321–345.
- Nattiez, J.-J. (1975). *Fondements d'une Sémiologie de la Musique*. Union Générale d'Éditions, Paris.
- Ostendorf, M., Digalakis, V., & Kimball, O. (1996). From HMMs to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Acoustics, Speech And Signal Processing*, 4, 360–378.
- Pachet, F. (2003). The continuator: Musical interaction with style. *Journal of New Music Research*, 32(3), 333–341.
- Pearce, M. (2005). The construction and evaluation of statistical models of melodic structure in music perception and cognition. PhD thesis, Department of Computing, City University, London.
- Pickens, J., & Crawford, T. (2002). Harmonic models for polyphonic music retrieval. In *Proceedings of the ACM International Conference on Information and Knowledge Management* (pp. 430–437). McLean, VA.
- Pienimäki, A., & Lemström, K. (2004). Clustering symbolic music using paradigmatic and surface level analyses. In *Proceedings of the Second Annual International Symposium on Music Information Retrieval* (pp. 262–265). Barcelona.
- Ponce de León, P., & Iñesta, J. (2003). Feature-driven recognition of music styles. In *Proceedings of the 1st Iberian Conference on Pattern Recognition and Image Analysis: Lecture Notes in Computer Science*, volume 2652 (pp. 773–781). Springer-Verlag.
- Ries, K., Buø, F., & Waibel, A. (1996). Class phrase models for language modeling. In *Proceedings of the International Conference on Spoken Language Processing* (pp. 398–401). Philadelphia.
- Rolland, P.-Y., & Ganascia, J.-G. (2000). Musical pattern extraction and similarity assessment. In Miranda, E. (Ed), *Readings in music and artificial intelligence*, chapter 7 (pp. 115–144). Amsterdam: Harwood Academic Publishers.
- Temperley, D. (2004). Bayesian models of musical structure and cognition. *Musicae Scientiae*, 8(2), 175–205.