Journal of Mathematics and Music, 2016
Vol. 10, No. 2, 127–148, http://dx.doi.org/10.1080/17459737.2016.1209588

Analysis of analysis: Using machine learning to evaluate the importance of music parameters for Schenkerian analysis

Phillip B. Kirlin[a*] and Jason Yust[b]

[a]*Department of Mathematics and Computer Science, Rhodes College, Memphis, USA;*
[b]*School of Music, Boston University, Boston, USA*

(*Received 16 November 2015; accepted 19 June 2016*)

While criteria for Schenkerian analysis have been much discussed, such discussions have generally not been informed by data. Kirlin [Kirlin, Phillip B., 2014 "A Probabilistic Model of Hierarchical Music Analysis." Ph.D. thesis, University of Massachusetts Amherst] has begun to fill this vacuum with a corpus of textbook Schenkerian analyses encoded using data structures suggested byYust [Yust, Jason, 2006 "Formal Models of Prolongation." Ph.D. thesis, University of Washington] and a machine learning algorithm based on this dataset that can produce analyses with a reasonable degree of accuracy. In this work, we examine what musical features (scale degree, harmony, metrical weight) are most significant in the performance of Kirlin's algorithm.

Keywords: Schenkerian analysis; machine learning; harmony; melody; rhythm; feature selection

2010 Mathematics Subject Classification: 68T05; 68T10
2012 Computing Classification Scheme: supervised learning; sound and music computing

1. Introduction

Schenkerian analysis is widely understood as central to the theory of tonal music. Yet, many of the most prominent voices in the field emphasize its status as an expert practice rather than as a theory. Burstein (2011, 116), for instance, argues for preferring a Schenkerian analysis "not because it demonstrates features that are objectively or intersubjectively present in the passage, but rather because I believe it encourages a plausible yet stimulating and exciting way of perceiving and performing [the] passage." Rothstein (1990, 298) explains an approach to Schenker pedagogy as follows.

> Analysis should lead to better hearing, better performing, and better thinking about music, not just to "correct" answers. [. . .] I spend lots of class time—as much as possible—debating the merits of alternate readings: not primarily their conformance with the theory, though that is discussed where appropriate, but their relative plausibility as models of the composition being analyzed.

Schachter (1990) illustrates alternative readings of many works and asserts that a full musical context is essential to evaluating them. Paring the music down to just aspects of harmony and voice leading, like "the endless formulas in white notes that disfigure so many harmony texts," he claims, leaves the difference between competing interpretations undecidable. In publications

*Corresponding author. Email: kirlinp@rhodes.edu

such deliberation typically occurs at a high level. It rarely addresses the implicit principles used to deal with many details of the musical surface. As Agawu (2009, 116) says, "the journey from strict counterpoint to free composition makes an illicit or – better – mysterious leap as it approaches its destination."

As with any complex human activity, the techniques of artificial intelligence may greatly advance our understanding of how Schenkerian analysis is performed and what kinds of implicit cognitive abilities and priorities support it. The present work builds upon the research of (Kirlin 2014a; 2015) which models Schenkerian analysis using machine learning techniques. By probing Kirlin's algorithm we address a question of deep interest to Schenkerian analysts and pedagogues: what roles do different aspects of the music play in deliberating between possible analyses of the same musical passage? Because the activity of Schenkerian analysis involves such a vast amount of implicit musical knowledge, it is treacherous to litigate this question by intuition, without the aid of computational models and methods.

This article is organized as follows. The next section, numbered 2, explains the machine learning algorithm we used, which is essentially that of Kirlin (2014b). Section 3 describes a series of experiments to test which musical features the algorithm relies upon most heavily to produce accurate analyses. Section 4 describes the results of that experiment, and Section 5 provides further exploratory analysis of data produced by the experiment.

2. A machine learning algorithm for Schenkerian analysis

Schenkerian theory is grounded in the idea that a tonal composition is organized as a hierarchical collection of *prolongations*, where a prolongation is defined, for our purposes, as an instance where a motion from one musical event, *L*, to another non-adjacent event, *R*, is understood to control the passage between *L* and *R*, and the intermediate events it contains. A prolongation is represented in Schenkerian notation as a slur or beam.

Consider Figure 1, a descending melodic passage outlining a G major chord. Assuming this melody takes place over G-major harmony, this passage contains two passing tones (non-harmonic tones in a melodic line that linearly connect two consonant notes via stepwise motion), the second note C and the fourth note A. These tones smoothly guide the melody between the chord tones D, B, and G. In Schenkerian terminology, the C *prolongs* the motion from the D to the B, and the A similarly *prolongs* the motion from the B to the G.

The hierarchical aspect of Schenkerian analysis comes into play when we consider a prolongation that occurs over the entire five-note passage. The slurs from D to B and B to G identify the C and A as passing tones. Another slur from D to G, which contains the smaller slurs, shows that the entire motion outlines the tonic triad from D down to G. The placement of slurs may reflect the relatively higher stability of the endpoints (between chord tones over non-chord tones, and between more stable members of the triad, root and fifth, over the third), or they may reflect a way in which the local motions (passing-tone figures) group into the most coherent larger-scale motion (arpeggiation of a triad).

Figure 1. A melodic line illustrating prolongations.

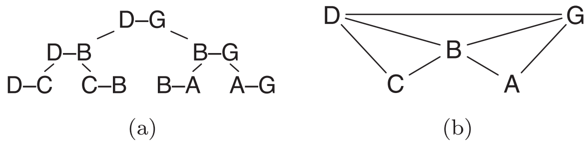

Figure 2. The prolongational hierarchy of a G-major chord with passing tones represented as two equivalent data structures. (a) A binary tree of melodic intervals. (b) A maximal outerplaner graph, or MOP.

This hierarchy can be represented visually by the tree in Figure 2(a): this diagram illustrates the hierarchy of melodic intervals present in the composition and the various prolongations identified above. An equivalent representation, known as a *maximal outerplanar graph*, or MOP (Yust 2006, 2009, 2015), is shown in Figure 2(b). Binary trees of intervals and MOPs are duals of each other in that they represent identical sets of information, though the MOP representation is more succinct.

From a mathematical perspective, a MOP is a complete triangulation of a polygon. Each triangle in a MOP represents a single melodic prolongation among the three notes of the music represented by the three endpoints of the triangle. Because MOPs are oriented temporally, with the notes shown in a MOP always ordered from left to right as they are in the musical score, we can unambiguously refer to the three endpoints of a triangle in a MOP as the left (L), middle (M), and right (R) endpoints. Each triangle in a MOP, therefore, represents a prolongation of the melodic interval from L to R by the intervals from L to M and M to R.

2.1. *A probabilistic interpretation of MOPs*

Our goal is to develop an algorithm with the ability to predict, given a musical composition, a "correct" Schenkerian analysis for that composition. We develop this algorithm using the following probabilistic perspective.

Assume that we are given a sequence of notes N that we wish to analyze, and that all possible Schenkerian analyses of N can be enumerated as $A_1, \ldots, A_m$, for some integer m. We desire the most probable analysis given the notes, which is $\arg\max_i P(A_i \mid N)$. Because a Schenkerian analysis A_i can be represented in MOP form by its collection of triangles, we will denote the set of triangles comprising analysis A_i by $T_{i,1}, \ldots, T_{i,p}$, for some integer p. We then define $P(A_i \mid N)$ as the joint probability $P(T_{i,1}, \ldots, T_{i,p})$. In other words, we define the probability of an analysis being correct for a given sequence of notes as the probability of observing a MOP containing the specific set of triangles derived from that analysis.

We will use supervised machine learning to estimate this probability from a corpus of Schenkerian analyses done by humans, which we interpret as ground truth. Specifically, we use the Schenker41 dataset, the largest known collection of machine-readable Schenkerian analyses in existence (Kirlin 2014a). This dataset contains 41 common-practice era musical excerpts and their corresponding Schenkerian analyses. All of the excerpts are either for a solo keyboard instrument (or arranged for such an instrument) or for voice with keyboard accompaniment. All are in major keys and do not modulate, though there are some tonicizations. The musical excerpts are encoded in the symbolic MusicXML format, whereas the corresponding analyses are encoded in a text-based format that captures the prolongational information in a Schenkerian analysis, while not assigning any musical interpretation to each prolongation. In other words, the prolongations are not labeled with terms such as "passing tone," "neighbor tone," or other descriptors, but are denoted only by their constituent notes. Each analysis also contains a Roman numeral harmonic labelling, determined either from the same source as the prolongations themselves, or from an expert music analyst.

The analyses in the dataset are derived from four sources: Forte and Gilbert's *Introduction to Schenkerian Analysis* and the accompanying solutions manual (1982a; 1982b) (10 excerpts), Cadwallader and Gagné's *Analysis of Tonal Music* (1998) (4 excerpts), Pankhurst's *Schenker-GUIDE* (2008) (24 excerpts), and teaching materials produced by an expert music analyst (3 excerpts).

Each excerpt in the dataset can be translated into a MOP representing the prolongations present in the main melody of the excerpt. However, it is difficult to estimate the full joint probability $P(T_{i,1}, \ldots, T_{i,p})$ directly from the resulting collection of MOPs owing to the curse of dimensionality: the large number of possible combinations of triangles is simply too large for any reasonably-sized dataset. Instead, we make the simplifying assumption that each triangle in a MOP is independent of all other triangles in a MOP, which implies $P(T_{i,1}, \ldots, T_{i,p}) = P(T_{i,1}) \cdots P(T_{i,p})$. This assumption reduces the full joint probability to a product of simpler, lower-dimensional probabilities, which are easier to learn. An experiment verifies that this assumption largely preserves relative probability scores between two MOPs, which is a sufficient condition for our purposes to proceed (Kirlin and Jensen 2011).

Finally, we define the probability of an individual triangle appearing in a MOP as the probability of a given melodic interval being elaborated by the specific choice of a certain child note. That is, we define $P(T_{i,j}) = P(M_{i,j} \mid L_{i,j}, R_{i,j})$, where $L_{i,j}$, $M_{i,j}$, and $R_{i,j}$ are the three endpoints (notes) of triangle $T_{i,j}$. In summary, we have defined the most likely analysis A_i for a given sequence of notes N as

$$\begin{aligned} \arg\max_i P(A_i \mid N) &= \arg\max_i P(T_{i,1}, \ldots, T_{i,p}) \\ &= \arg\max_i \prod_{j=1}^{p} P(T_{i,j}) \\ &= \arg\max_i \prod_{j=1}^{p} P(M_{i,j} \mid L_{i,j}, R_{i,j}). \end{aligned} \tag{1}$$

We now return to the question of using the Schenker41 corpus to compute the probability $P(M_{i,j} \mid L_{i,j}, R_{i,j})$, which requires us to define exactly what type of musical information underlies the variables $L_{i,j}$, $M_{i,j}$, and $R_{i,j}$. It is typical to describe these variables by a collection of *features*: specific measurable properties of the source music whose values are determined from the notes of the music corresponding to those variables. We use a set of 18 features that provide basic melodic, harmonic, metrical, and temporal information about the music; these are described more specifically in Section 3.

With a set of features in hand, a straightforward approach to determining the probability above involves counting the frequency of every type of triangle within the corpus, where each type of triangle is determined from a unique combination of the 18 features. The curse of dimensionality thwarts us again: there are too many possible types of triangle to expect to get a reasonable frequency count from any reasonably-sized corpus of analyses. Instead, we use *random forests* (Breiman 2001), a machine learning ensemble method particularly suited for high-dimensional feature spaces, to learn the conditional probability $P(M_{i,j} \mid L_{i,j}, R_{i,j})$. Random forests, which operate by constructing a large collection of decision trees, are typically used for classification tasks by outputting the most frequent class (the mode) predicted by the collection of trees. However, a probability distribution can be obtained instead by counting the frequencies of the predicted classes and normalizing them (Provost and Domingos 2003).

Our set of 18 features includes a subset of six features that depend on the middle note $M_{i,j}$. It is unlikely that a single random forest could sufficiently learn to predict all six features simultaneously. Therefore, we factor our conditional probability into the product of six different probabilities, each assigned to predict one of the six features of the middle note. If we denote these six features of the middle note as M_1 through M_6, this factorization becomes (dropping the i, j subscripts for brevity):

$$\begin{aligned} P(M \mid L,R) &= P(M_1, M_2, \ldots, M_6 \mid L,R) \\ &= P(M_1 \mid L,R) \cdot P(M_2, M_3, \ldots, M_6 \mid M_1, L, R) \\ &= P(M_1 \mid L,R) \cdot P(M_2 \mid M_1, L, R) \cdot P(M_3, M_4, \ldots, M_6 \mid M_1, M_2, L, R) \\ &= P(M_1 \mid L,R) \cdot P(M_2 \mid M_1, L, R) \cdots P(M_6 \mid M_1, \ldots, M_5, L, R). \end{aligned}$$

In other words, we construct six random forests and use each one to construct a conditional probability distribution over a single feature of the middle note. Multiplying these distributions together gives us a distribution over all six features of the middle note, and therefore an estimate of the probability of seeing any particular triangle in a MOP.

Now that we have an appropriate estimate of this probability $P(M_{i,j} \mid L_{i,j}, R_{i,j})$, we can calculate the probability of an entire MOP analysis according to equation (1). Computationally, it is inefficient to enumerate all possible Schenkerian analyses for a given sequence of notes; the size of this set grows exponentially with the length of the sequence. Instead, by combining the probabilistic interpretation of MOPs along with the equivalence between MOPs and binary trees, we can view each prolongation within a MOP as a production in a probabilistic context-free grammar. Under this interpretation, it is straightforward to use standard parsing techniques (Jiménez and Marzal 2000; Jurafsky and Martin 2009) to develop an $O(n^3)$ algorithm, known as ParseMop-C, that determines the most probable analysis for a given sequence of notes. Additionally, the grammar formalism allows us to restrict the predicted analyses to those that contain a valid *Urlinie* through specific sets of production rules.

ParseMop-C accepts as input a sequence of notes to analyze, along with information about the harmonic context of those notes and what category of *Urlinie* should be located (e.g., $\hat{5}$-line or $\hat{3}$-line). The algorithm begins by computing the probabilities of analyses of all three-note excerpts of the input music. Because a three-note excerpt forms exactly one triangle in a MOP, these probabilities come directly from the output of the random forests. The most probable analyses for all four-note excerpts are obtained by combining the pre-computed results for three-note excerpts. This continues with n-note excerpts being analyzed by combining probabilities of $(n-1)$-note excerpts, until the entire piece is analyzed. The most probable analysis is output as a MOP, though there is an additional algorithm available that can display the analysis in more traditional Schenkerian notation.

We can compare the predicted output analyses of ParseMop-C against the ground-truth analyses from the Schenker41 corpus to determine the performance of the algorithm, which we measure via *edge accuracy*, which is the percentage of edges in an algorithmically-produced MOP that correspond to edges in the ground-truth MOP. Additional details on the probabilistic interpretation of MOPs, ParseMop-C, and its evaluation are available in Kirlin and Jensen (2015) and Kirlin (2014b).

ParseMop-C is the first completely data-driven Schenkerian analysis algorithm, in that it learns all the “rules” of Schenkerian analysis, including their forms and where the apply them, completely from a corpus of data. Very early studies in computional approaches to Schenkerian analysis encountered difficulties due to the issues inherent in handling the seemingly conflicting guidelines of the analysis process (Kassler 1975, 1987; Frankel, Rosenschein, and Smoliar 1978). Lerdahl and Jackendoff’s *A Generative Theory of Tonal Music* (1983) presented two different formal systems for music that, like Schenkerian analysis, can be used to view a piece of music as a hierarchy of musical objects. The authors, however, stated that their formalizations were not designed to replicate the ideas of Schenkerian analysis, but were rather a new investigation in musical cognition. Though Lerdahl and Jackendoff's systems lack the details necessary to develop a "computable procedure for determining musical analyses," a number of further endeavors attempted to fill in the gaps. The most successful of these is the work done by Hamanaka et al., which has resulted in a collection of computational systems that attempt to replicate parts of Lerdahl and Jackendoff's theories. Their early systems (Hamanaka, Hirata, and Tojo 2005, 2006, 2007) required user-supplied parameters to identify the optimal music analysis, while later systems (Hamanaka and Tojo 2009; Miura et al. 2009; Kanamori, Hamanaka, and Hoshino 2014) have focused on using machine learning to automate the search for the appropriate parameters. Numerous other approaches (Mavromatis and Brown 2004; Gilbert and Conklin 2007; Marsden 2005; Marsden and Wiggins 2008; Marsden 2010) have used a context-free grammar formalization similar to the one presented above, but with a hand-created rule set rather than rules learned directly from the ground-truth analyses.

3. Musical features

To understand how individual musical features might figure into the accuracy of these algorithmic analyses, consider the task of analyzing the "music" in Figure 3. This is the intervallic pattern of the melody of a simple four-measure phrase of real music. There is no information about the rhythm or harmony of the original music. We do not know what the key is or which note is the tonic. Registral information is also removed by octave-reducing melodic intervals and inverting larger leaps. How likely would it be that one could reproduce a textbook analysis of the melody relying on this intervallic pattern only?

Or what if one were given only the rhythm and meter of a melody, as in Figure 4? How accurately could we expect to predict a Schenkerian analysis of the passage?

In the experiment described below, we train the machine learning algorithm on Schenkerian analyses with differing amounts of information about the music, and therefore discover how important different aspects of the music (melodic, harmonic, metric) are to accurately reproducing Schenkerian analyses.

The data for Figures 3 and 4 comes from Schubert's Impromptu Op. 142, No. 2, which is analyzed by Cadwallader and Gagné (1998) as shown in Figure 5. Their analysis has enough detail to be translated into the explicit and near-complete MOP shown in Figure 6(**a**). The MOP is a simplification, reflecting just the basic melodic hierarchy implied by Cadwallader and Gagné's analysis. There are three places where their analysis is ambiguous: in measures 1–3, they show a double neighbor figure A♭-G-B♭-A♭, where a slur could be added from A to B♭ or from G to A♭, but neither is clearly implied by the analysis. In measures 5–8, they show C-B♭-A♭-C in stemmed notes. A slur from C to A♭ might be inferred but is not actually shown, so it is not included in the coded analysis. Finally, this example reflects a common problem in that Cadwallader and Gagné give the status of $\hat{3}$ to two different Cs. A fully explicit analysis would choose between

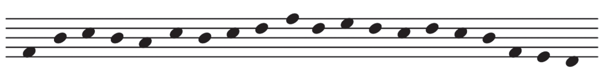

Figure 3. A melody with general interval information only.

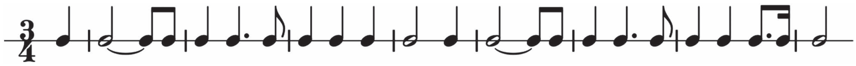

Figure 4. The meter and rhythm of a melody.

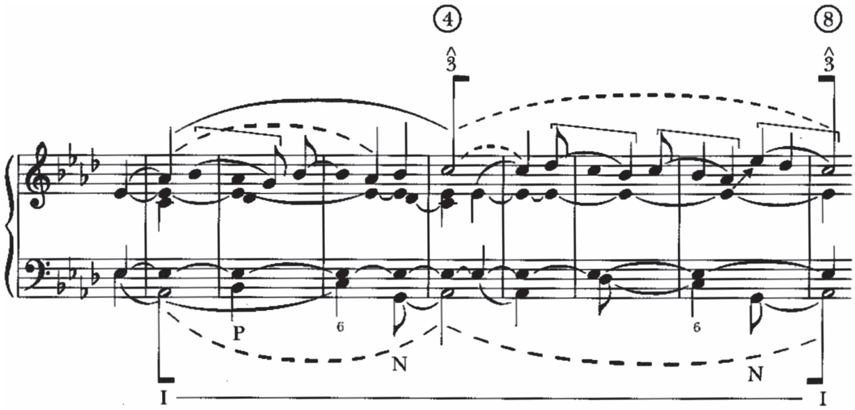

Figure 5. Cadwallader and Gagné (1998)'s analysis of Schubert's Impromptu Op. 142, No. 2.

Figure 6. (**a**) An encoded version of the analysis in Figure 5; (**b**) an analysis of just the intervallic pattern of the impromptu, as shown in Figure 3, by a computer trained on just the intervallic patterns of textbook analyses; (**c**) computer analysis of the melodic (interval and scale degree) and harmonic information; and (**d**) a computer analysis of the melodic, harmonic, and metrical data of the impromptu.

these as the true initiation of the fundamental line. The part of the analysis shown with beams is the "background" of the analysis; the ParseMop-C algorithm is constrained to find such a $\hat{1}$-$\hat{2}$-$\hat{3}$ initial ascent as the background of the phrase (but not to find these notes in any specific locations).

Figures 6(**b**)–6(**d**) show three analyses by the algorithm trained on limited amounts of musical information, which can be compared to the "ground truth" in Figure 6(**a**). The first, Figure 6(**b**), shows the solution arrived at by the algorithm trained on just the generic melodic intervals of the corpus of textbook analyses. Since this version of the algorithm has no awareness of keys, accidentals, harmony, register, or rhythm, the music is shown with no key signature, clef, or durational values, and with octave-reduced generic intervals (as in Figure 3). Note that adjacent notes are implicitly connected, so a slur between the first two notes would be redundant. The algorithm accurately identifies some local passing motions in the music given just this basic generic intervallic information. Evaluated by the proportion of shared slurs between the computer analysis and the textbook one, however, the computer does not do especially well on this example, mostly because it buries the C of measure 4 in the middle of a B♭-C-D♭ passing motion, and thus most of its slurs cross over a note that is especially prominent in the textbook analysis. This passing motion, which is perfectly reasonable given only intervallic information, is quite implausible when we see the metric and harmonic status of the three notes involved. On other examples, however, the algorithm performs surprisingly well with such minimal musical information, as we shall see below.

Given a slightly richer musical object, including the scale-degree number of each note (i.e., it has the reference point of a key), and some harmonic context (Roman numeral without inversion) the algorithm produces the analysis of Figure 6(**c**). The new information allows it to avoid certain blatant errors: for instance, now that it knows that the third note (B♭) occurs over a tonic chord, not a V, it avoids assigning it a major structural role. However, the algorithm makes another decision that turns out to be a mistake, shifting the first note of the structure ahead to note 7. It apparently identifies an arpeggiation of V as a likely introductory approach to the first structural note, but that turns out to be implausible because of the rhythm. When it has additional metrical information (specifically, which notes are accented relative to others), the algorithm corrects this mistake, shifting the structural beginning back to the metrically strong note 2, as shown in Figure 6(**d**). This result is then quite similar to Cadwallader and Gagné's analysis.

This single example shows how different aspects of the music – melodic pattern, harmony, rhythm – play different roles in determining the plausibility of a Schenkerian analysis. The principal goal of the experiment reported here is to answer the broad question of *what features of the music are most essential to accurately reproducing human analyses.* To clarify, this experiment is not designed to select a "best" subset of variables that will be used in future studies; using random forests as our learning algorithm already reduces much of the need to find an ideal subset of variables in order to simplify the model or combat overfitting (Hastie, Tibshirani, and Friedman 2009). The motivations of this study come, instead, primarily from music theory. For instance, theorists often emphasize the linear aspect of Schenkerian analysis, to distinguish it from other analytical methods that focus more heavily on chords and root progressions. Without a data-driven approach, however, it is not really clear how much a Schenkerian analysis really relies on melodic aspects of the music as opposed to harmonic progressions. Also, Schenker pedagogy typically de-emphasizes the metrical aspect of music, by illustrating, for instance, how brief and metrically weak notes may sometimes play central structural roles. Yet, because analysts may rely upon metrical information unconsciously, in making less prominent analytical decisions, such arguments may be misleading. Finally, Schenkerian analysis is often tacitly understood to be a recursive procedure, meaning that the same criteria apply to very local analytical decisions as those at higher levels, between non-adjacent notes and measure-to-measure across a phrase. Although the examples in the Schenker41 corpus are primarily short – usually just a single phrase or pair of phrases – we were able to test a limited version of this hypothesis from the note-to-note to the measure-to-measure level, by including a set of temporal features, showing that even this limited version of the recursion hypothesis is incorrect.

The main results (reported in Section 4) are the outcomes of two feature selection processes. The first tests the relative importance of broadly-categorized harmonic, melodic, metric, and temporal features. The second is an ordering of the more specific set of 18 features according to how much the performance of the machine learning algorithm depends upon them. This second process generated a substantial amount of data that also permitted us to draw some limited conclusions via exploratory analysis of how features interact. This is reported in Section 5.

The features used in this work were originally specified by Kirlin (2014b) because they covered a range of types of musical information, melodic, harmonic, rhythmic, and temporal, were easily determinable by computer from the dataset, had a small number of possible values for each feature, and most importantly, seemed likely to fulfill a critical role in the Schenkerian analytical process. The features also proved well-suited to test the kinds of common assumptions about Schenkerian analysis described above.

We define 18 features in all that are available to the algorithm in the full model. The following six of these are features of the middle note.

- **SD-M** The scale degree of the note (represented as an integer from 1 through 7, qualified as raised or lowered for altered scale degrees).
- **RN-M** The harmony present in the music at the time of onset of the center note (represented as a Roman numeral from I through VII or "cadential six-four"). For applied chords (tonicizations), labels correspond to the key of the tonicization.
- **HC-M** The category of harmony present in the music at the time of the center note, represented as a selection from the set *tonic* (any I chord), *dominant* (any V or VII chord), *predominant* (II, II^6, or IV), *applied dominant*, or *VI chord*. (Our dataset did not have any III chords.)
- **CT-M** Whether the note is a chord tone in the harmony present at the time (represented as a selection from the set "basic chord member" (root, third, or fifth), "seventh of the chord," or "not in the chord").
- **Met-LMR*** The metrical strength of the middle note's position as compared to the metrical strength of note *L*, and to the metrical strength of note *R* (represented as a selection from the set "weaker," "same," or "stronger").
- **Int-LMR*** The melodic intervals from *L* to *M* and from *M* to *R*, generic (scale-step values) and octave generalized (ranging from a unison to a seventh).

Two features of the middle note are different than the others, in that they are influenced by the left and right notes. These are therefore distinguished as "LMR" features, as opposed to simple "M" features.

We also used the following 12 features for the left and right notes, *L* and *R*.

- **SD-LR*** The scale degree (1–7, qualified as in SD-M) of the notes *L* and *R*.
- **Int-LR** The melodic interval from *L* to *R*, with intervening octaves removed.
- **IntI-LR** The melodic interval from *L* to *R*, with intervening octaves removed and intervals larger than a fourth inverted.
- **IntD-LR** The direction of the melodic interval from *L* to *R*; i.e., up or down.
- **RN-LR*** The harmony present in the music at the time of *L* or *R*, represented as a Roman numeral I through VII, or "cadential six-four."
- **HC-LR*** The category of harmony present in the music at the time of *L* or *R*, represented as a selection from the set tonic, dominant, predominant, applied dominant, or VI chord.
- **CT-LR*** Status of *L* or *R* as a chord tone in the harmony present at the time (see the description of CT-M above).
- **MetN-LR*** A number indicating the beat strength of the metrical position of *L* or *R*. The downbeat of a measure is 0. For duple or quadruple meters, the halfway point of the measure is 1; for triple meters, beats two and three are 1. This pattern continues with strength levels of 2, 3, and so on.
- **MetO-LR*** A number indicating the beat strength of the metrical position of *L* or *R*. The downbeat of a measure is 0. For duple or quadruple meters, the halfway point of the measure is 1; for triple meters, beats two and three are 1. This pattern continues with strength levels of 2, 3, and so on. The difference between MetN-LR and MetO-LR is that the former is represented in trees of the random forest as a *numeric* variable, and the latter as an *ordinal* variable.

The algorithm that creates the forests treats these types of variables differently: numeric variables are compared using greater-than or less-than tests, while ordinal variables are treated as categorical, and are compared using equality tests.

- **Lev1-LR** Whether *L*, *M*, and *R* are consecutive notes in the music, represented as a true/false value (this can be determined strictly from examining the positions of *L* and *R*, and so is not included in the list of middle-note features).
- **Lev2-LR** Whether *L* and *R* are in the same measure in the music, represented as a true/false value.
- **Lev3-LR** Whether *L* and *R* are in consecutive measures in the music, represented as a true/false value.

Note: Features marked with an asterisk (*) *in the lists above are, from a machine learning standpoint, two different features: they are separate and distinct inputs available to the random forests algorithm; for instance, there is an SD-L feature and an SD-R feature that the learning algorithm treats independently. However, in the feature selection experiments that follow, these "paired features" are always treated as an indivisible unit in the set of available features– they are always added or removed together.*

There are many well-established feature selection techniques that could potentially help determine the relative importance of the features in producing output analyses that match the ground-truth analyses. Many are inappropriate in our situation, however, because they are designed specifically for classification tasks, and there is no notion of an output class in our system. A further complication is that any change in the input features directly affects the output of the random forests, which is a probability distribution for which we do not have ground-truth data. Certainly, improving the estimate of the probability distribution will likely improve the quality of the most-probable music analysis predicted by ParseMop-C, but it is unclear to what degree this is true, as the two algorithms are computationally separate. Therefore, we turned to two standard subset selection methods along with examining correlations between features.

In the two experiments that follow, we evaluated the quality of a specific subset of features by running the ParseMop-C algorithm with leave-one-out cross-validation to compute new MOP analyses for all the pieces in the Schenker41 corpus longer than four measures, then calculated the overall edge accuracy for all the MOPs combined. For our first experiment, we divided the features into four broad categories, melodic, harmonic, rhythmic, and temporal, and evaluated the quality of all exhaustive subsets of these categories of features. In our second experiment, we used *backward selection* to cycle through each feature from the training data and in order to find the feature that, when omitted, decreased the overall accuracy the least. This feature was then permanently removed and another cycle was performed, until only one feature remained. For the entire set of 18 features we then had 18 trials, numbered 0–17, where trial 0 included all 18 features in the full model, and trial 17 (trivially) included only the last remaining feature.

4. Results

4.1. *Experiment 1*

We first divided the features into four categories as follows.

- Melodic: SD-M, SD-LR, Int-LMR, Int-LR, IntI-LR, IntD-LR
- Harmonic: RN-M, RN-LR, HC-M, HC-LR, CT-M, CT-LR
- Metrical: Met-LMR, MetN-LR, MetO-LR
- Temporal: Lev1-LR, Lev2-LR, Lev3-LR

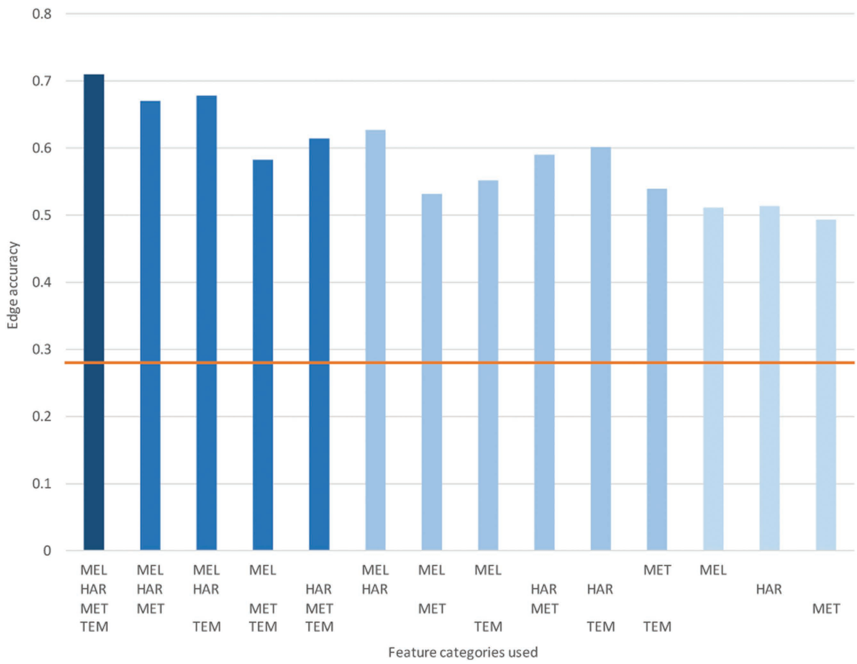

Figure 7. Edge accuracy as determined from different subsets of feature categories. There is no trial for the temporal features by themselves, as there are no middle-note features in that category, and at least one such feature is required. The horizontal line at 28% indicates the baseline edge accuracy that could be obtained by an algorithm that operated randomly.

We ran 14 trials to evaluate the quality of non-zero sized subsets of feature categories, from all four categories (including all 18 features) down to single categories. Each trial produced an overall edge accuracy giving us information on the utility of the categories of features, as detailed in Figure 7. The random forests use the left–right features as "input" and the middle-note features as "output," and therefore trials are impossible to run if no middle-note features are included. Thus, there is no "temporal-only" trial.

Figure 7 shows a baseline edge accuracy of roughly 28%. This number is calculated by averaging the edge accuracies of all possible MOP analyses of all the musical excerpts in question. In other words, a hypothetical version of ParseMop that analyzed music by selecting a MOP analysis uniformly at random from all possibilities would average 28% edge accuracy.

The data from this experiment allow us to test three broad music-theoretic questions described in Section 3: first, what is the relative importance of melodic and harmonic information? Second, is metrical information necessary for producing accurate analyses? Third, does temporal information influence analyses – that is, is it necessary to use harmonic, melodic, and metrical information differently depending on whether one is making decisions at note-to-note or measure-to-measure levels? All of these questions essentially presume that each feature set has a relatively consistent influence over the performance between trials, regardless of the other feature sets present. Figure 8 therefore shows the change of performance between trials that differed only in the presence or absence of one feature set, melodic, harmonic, metrical, or temporal. These appear to reflect relatively normal distributions, so we further assessed the contribution of each feature set performing paired t-tests comparing data with and without the given feature, and found the inclusion of harmonic, melodic, and temporal features made a significant difference at $p < .001$ and metrical features at $p < .01$.

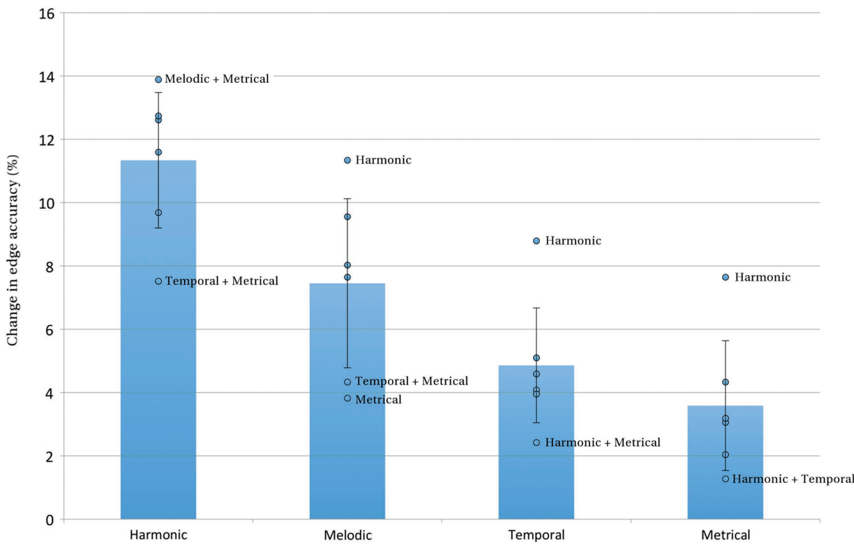

Figure 8. Change of performance for adding each feature set. Each data point is the difference in performance between trials that include the feature set listed on the *x*-axis and trials with those features removed. Averages and standard deviations are shown for the addition of each group of features. Data points falling outside of a single standard deviation of the average are labeled with the other feature sets present on the trials compared. For instance, the largest difference observed is between the trial with Melodic + Metrical features, and the one with Melodic + Metrical + Harmonic features.

From Experiment 1 we can draw the following conclusions.

- Harmonic features are the most essential to producing an accurate analysis. In particular, they outweigh melodic features, although these also are very important. This is shown in the large average influence over performance exhibited by these feature groups in Figure 8.
- Temporal features have a reliably positive influence on performance, around 5% on average in Figure 8, meaning that the rules of Schenkerian analysis do differ from level to level. In particular, this influence seems to be present regardless of what other feature sets are present, so we can infer that harmonic, melodic, and metrical information all need to be used differently in some way between note-to-note, within-measure, and/or measure-to-measure levels.
- Metrical features also show a reliable influence over performance in Figure 8, although it is relatively small (around 3–4%).

The extreme values in Figure 8 are also suggestive. All feature sets have well above-average influence when just the harmonic features are present, which suggests that harmonic features are most useful in combination with at least one other feature type. At the same time, three of the unusually low values point to the fact that one trial, Harmonic + Temporal + Metrical, is substantially lower than what would be predicted by a simple additive model of how the features interact. In other words, there appears to be an especially sizeable overlap in analytically usable information for this particular combination of features.

4.2. *Experiment 2*

The second experiment compared the expendability of the 18 individual features using the backward selection procedure. Figure 9 lists the features dropped on each trial. At certain points in the process there are larger changes in baseline edge accuracy, most notably on trials 14, 16, and 17.

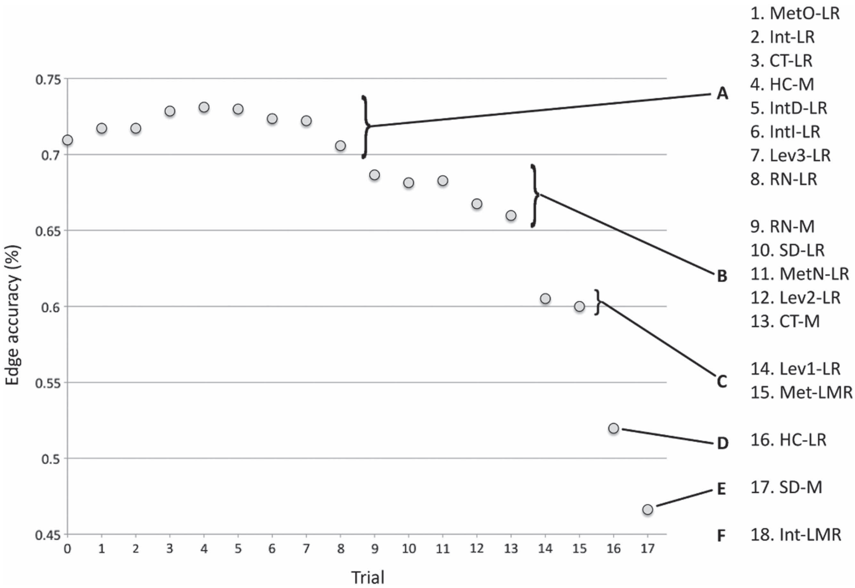

Figure 9. Performance of the algorithm on trials 0–17 and the feature removed prior to each trial. (Trial 0 includes all features.) N.B. There is no trial 18, but Int-LMR is listed as number 18 to show that it is the last feature remaining in the model.

This suggests that between these points are more significant differences in feature importance, and the features can be sorted into groups between these critical points.

As a check on the robustness of the ordering produced by the backward selection we also averaged, across all trials including the given feature, the decrease in performance observed after removing that feature. These data are shown in Figure 10. The orderings are consistent between the groupings indicated on Figure 9, but not within the larger groups A and B.[1] We also performed paired *t*-tests on the data comparing trials that differed only on the inclusion of the given feature. The results are shown in Figure 10. Note that *n* decreases as one goes higher on the list in Figure 9, and is very small for many of the features in group A. Nonetheless, the statistical tests show reliable influence over performance for all of the features in groups B–F, and even for some features in group A.

One striking aspect of the result is that, if we sort the features by type – melodic, harmonic, metrical, and temporal – the five features in groups C–F represent all four types, with a duplication for melodic features (Int-LMR and SD-M). In addition, the five features in group B also represent all types, with an additional harmonic feature (CT-M and RN-M). Since the ordering of groups C–F appears robust, while that within group B is not, we can broadly characterize the results as follows.

- The result of Experiment 2 is consistent with the finding from Experiment 1 that melodic, harmonic, metrical, and temporal information all contribute to the analytical process. It also indicates, further, that the algorithm performs best with a mix of all these feature types.
- It is also consistent with the finding from Experiment 1 that melodic and harmonic features are the most important (groups D–F), metrical and temporal somewhat less so (group C).

[1] These two measures are confounded: for the features that remain in the model for longer, there are more trials to average over, and the drop in performance predictably gets higher as the number of features in the full model gets smaller. The similarity of the two rankings can therefore be partly featured to this confound, but not entirely so. The average rise of feature importance from trial to trial is just 0.12%, and this is mostly due to the last four trials. On trials 0–13 the average change is 0.02%.

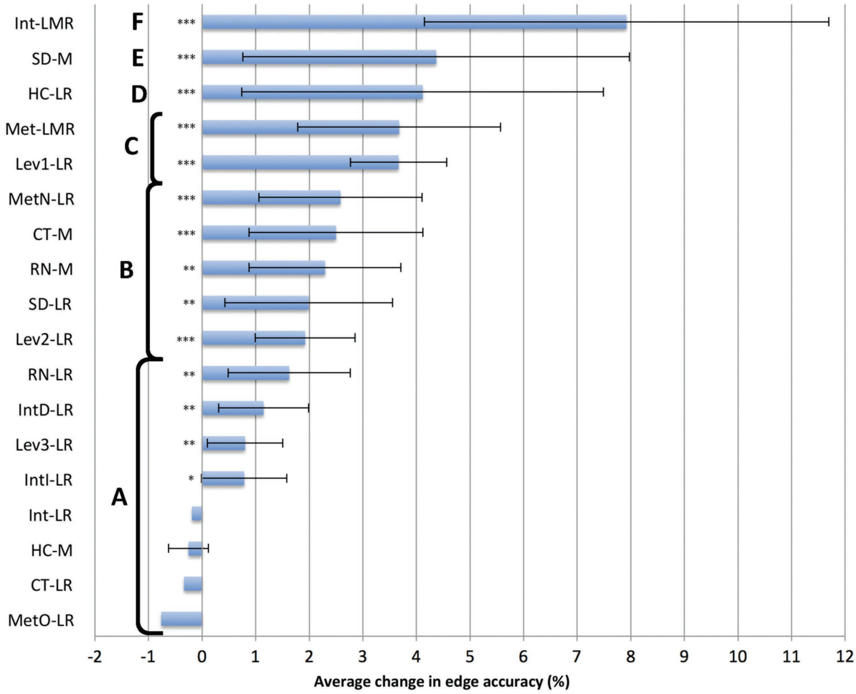

Figure 10. Average feature importance and standard deviation for all features over the course of the experiment. Asterisks show the results of paired t-tests between trials including and excluding the given feature. $^{***} = p < .001$, $^{**} = p < .01$, $^{*} = p < .05$.

- For each of the feature types, except melodic, the majority of the necessary information is captured by a single feature. Yet, in all cases, some additional valuable information is provided by one or two other features (group B). In the case of melodic features, these can be subdivided into purely intervallic information (Int-LMR) and tonally anchored information (SD-M).

While the results of the two experiments are largely consistent, they give divergent indications on the relative significance of melodic and harmonic features. Experiment 1 indicated that harmonic features make a larger contribution to the analytical process than melodic ones, while in experiment 2, the highest-ranking features are melodic ones (Int-LMR and SD-M). Together, these two results indicate that the important melodic information is consolidated in these two specific features, which provide a melodic pattern (Int-LMR) and orient it to a tonal center (SD-M), whereas harmonic information is distributed among several different kinds of features: distinct RN/HC versus CT features, the former dividing up between M and LR types.

Other than SD-M and Int-LMR, the only other notable melodic feature is SD-LR (group B). For the most part, the scale degree of the left and right notes is predictable from the scale degree of the middle note and the intervals to and from the middle note. While SD-LR provides an additional distinction between chromatically altered scale degrees and diatonic ones, the significance of this feature probably is more due to the potential use of melodic information as an LR feature – i.e. on the left-hand side of the conditional probabilities. This is further discussed in Section 5.

Three of the harmony features drop out in group A. The HC-M and RN-M features, and HC-LR and RN-LR, are highly redundant (see Section 4.4 below), so it is unsurprising that one of each of these drops out in group A. The early exclusion of the CT-LR feature is also unsurprising, because non-chord-tones should be rare as LR features in the corpus. The other three harmony features all seem to be important: HC-LR, CT-M, and RN-M. HC-LR is the last LR feature (group D), and from trial 11 onward the only other LR features are temporal ones. This indicates that harmony provides the most useful context for predicting features of the middle note. CT-M is also a feature that would be expected to be important, especially at the note-to-note level, since non-chord-tones are usually afforded a low structural status. Some additional data analysis reported in Section 5 suggests that on trial 13, after the removal of CT-M, a group of other features (SD-M, HC-LR, Int-LMR, and MetN-LR) combine to predict the chord-tone status of the middle note effectively.

As in Experiment 1, the metrical and temporal features played a lesser but nonetheless significant role. Of the temporal features, both Lev1-LR and Lev2-LR were valuable, showing that it is mainly the note-to-note, within-measure, and between-measure levels that behave differently. The difference between consecutive measures and non-consecutive measures (indexed by Lev3-LR) was less useful. However, this may simply reflect the fact that the musical examples were relatively short, and important decisions at these higher levels were constrained by the given fundamental line.

Between the metrical features, it is unsurprising that Met-LMR is especially useful, since it incorporates relative information about the left and right notes as they relate to the middle note. However, MetN-LR, which provides absolute information about the metrical context, is also of some value.

4.3. *Experiment 3*

To complement the information about the relative expendability of features provided by the backward selection process in Experiment 2, we also performed a more limited forward selection process to assess the independent usefulness of smaller sets of features. Because M features require LR features, and vice versa, we paired all similar M and LR features, and also included the LMR features by themselves. The results are shown in Figure 11.

These results corroborated the usefulness of harmony, observed in Experiment 1 but less evident in Experiment 2. Melodic features alone give lower accuracy than either harmonic (not including CT) or metrical. This suggests that the high analytical value of melodic information is dependent upon some baseline harmonic and metrical information (in the form of HC-LR and Met-LMR), while harmony and meter are somewhat more independent. The especially low performance of the CT features can be attributed to the uselessness of CT-LR (given that the great majority of triangles in the corpus will have chord tones on the left and right), also observed in Experiment 2.

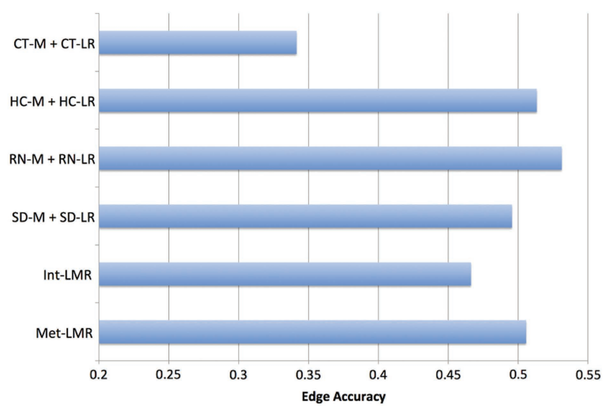

Figure 11. The edge accuracy of four models containing only a single LMR feature, or a single M and LR feature.

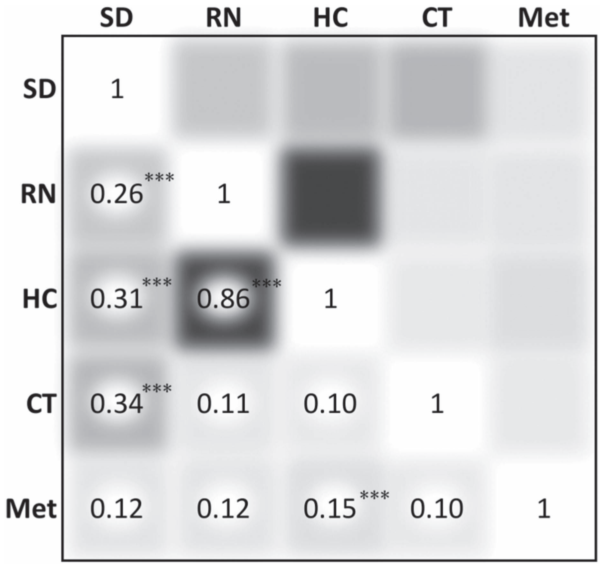

Figure 12. Cramer's V for five musical features (SD = scale degree, HC = harmony class, RN = Roman numeral, CT = chord tone status, and MS = metric strength). The numerical values are given below the diagonal and also visually illustrated by the relative darkness of squares in the grid. *** = $p < .001$.

4.4. *Correlations among features*

To aid in understanding how features may be interacting in the two main experiments, we also calculated correlations between features that existed in the musical excerpts themselves, without regard to the analyses (Figure 12). Having two features highly correlated with one another included in the model can be expected to reduce the overall influence of both features. Thus, knowing which features are correlated in the music can help to separate the influence of these correlations from redundancies that are specific to the analytical process.

The five features included are all those that can apply to a single note: distinctions having to do with relationships between notes are not included. There is one melodic feature (scale degree), three harmonic features (harmonic class, Roman numeral, and chord tone), and one metrical feature (metric strength). A number of correlations are significant at $p < .001$ (and no others were significant even at $p < .05$). Metric strength has no significant correlations except for a small one with harmony class. On the other hand, melodic information is strongly correlated with all of the harmonic features. This is unsurprising for RN and HC, but the high value for CT is more so. This high correlation is due primarily to the strong tendency of the "7th of chord" category to be represented by $\hat{4}$. Among the harmonic features, the very high correlation between HC and RN is unsurprising. These features differ mainly on the treatment of applied chords, which are rare, and otherwise on the greater specificity of the RN feature. CT is not significantly correlated with the other two harmonic features.

5. Interactions between features

The results in the previous section give a general picture of what kind of musical information is most important in producing a Schenkerian analysis. However, we also found that the importance of a given feature may be dependent upon what other features are present in the model. In some cases, the reasons for this are obvious. For instance, we found in Section 4.4 that scale degree and harmony class are strongly correlated in the music. Therefore, the presence of, e.g., SD-M should reduce the importance of HC-M (and SD-LR should reduce the importance of HC-LR, etc.). Where this reflects features of the music, it has nothing to do with Schenkerian analytical practice *per se*. However, similar kinds of redundancies might derive from the nature of the analytical process as well. And we also find that certain features *increase* in importance when another feature is included, which reflects an interdependence of features in the analytical process.

The backward selection process, because it required removing every remaining feature on every trial, produced data that made it possible to address this question of how features interact. The nature of this data analysis is necessarily exploratory. Consider, for instance, Figure 13, which tracks the effect of removing the scale degree feature over the course of Experiment 2. Were the relationship of this to other features equivocal, we might expect the values to stay roughly the same or to increase steadily from one trial to the next. The chart for SD-M is much more jagged, suggesting that it has non-trivial relationships with other features. For instance, when SD-LR is removed, SD-M suddenly spikes in importance. This suggests that the preceding low value probably reflected not a lack of significance for scale degree information in general, but that there is considerable overlap between the two scale degree features, making one of them – but not both – expendable. This makes sense, because given that the algorithm has intervallic information about the melody (in the form of Int-LMR), either scale degree feature can anchor that intervallic information to a tonal center.

The potentially most meaningful data in these graphs are the places like this where there are large changes in the importance of some feature from one trial to the next. We can infer that these large changes result from some kind of interaction between the feature in question and the one removed from one trial to the next. These may be of the form of redundancies or interdependencies. Such redundancies or interdependencies might involve more than one feature. For example, if one knows the scale degree of the middle note (SD-M) then one can infer the scale degree of the left and right notes (SD-LR) by knowing the intervals between the middle and the left and right notes (Int-LMR). Therefore, these three features are redundant as a group, but no pair of them is redundant.

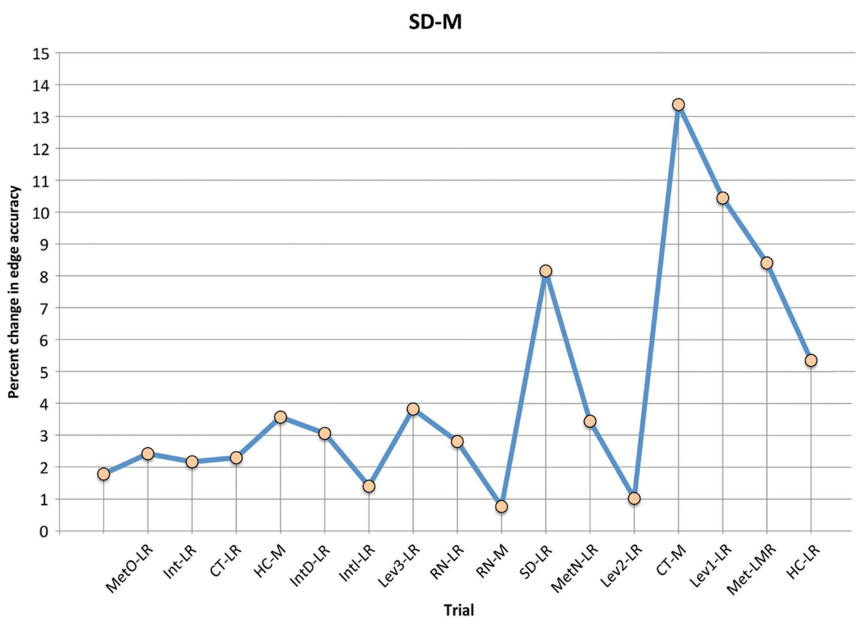

Figure 13. The effect of removing the SD-M feature on trials 0–16. The labels on the horizontal axis show the feature removed prior to the given trial. For instance, the peak at the trial for SD-LR means that the effect of removing SD-M changed to about 8% *after* the removal of SD-LR.

This redundancy is reflected by spikes in the importance of SD-M in Figure 13. The drops in importance reflect interdependencies between features. The largest drop in Figure 13 occurs when MetN-LR is removed. This indicates that SD-M is more useful when MetN-LR is included in the model. MetN-LR provides absolute metric orientation to complement the relative information of Met-LMR, so we can deduce that the likelihood of a given scale degree in the middle position changes depending on the metric context. We also see fairly large drops in the importance of SD-M for the temporal features Lev1-LR and Lev2-LR, which similarly indicates that the likelihood of certain scale-degree progressions depends on temporal context.

To search the data systematically for the most prominent such interactions between parameters, we tallied every change of feature importance from one trial to the next and ordered them from largest to smallest in distance from the mean (where "feature importance" is the reduction of triangle accuracy that results from removing the given feature on the given trial). The mean change is 0.12%, indicating a tendency for feature importance to increase modestly on average as the model gets smaller. Table 1 includes all cases where the change is further than one standard deviation (2.35%) from this mean.

We will not address every entry in this list. The sections below will instead discuss some of the more notable interactions shown.

5.1. *Removal of CT-M*

The largest two changes, and four of the largest 20, occurred between trials 12 and 13 after the removal of the CT-M feature. All of these are increases in sensitivity, which indicate redundancies. These redundancies apparently involve a large group of parameters. Most non-chord-tones will occur when the left and right notes are of the same harmonic class and the middle note has a conflicting scale degree. Therefore, the combination of HC-LR and SD-M can substitute for CT-M in many instances, making for a three-way redundancy. This is indicated in the first two entries of Table 1.

Other kinds of non-chord-tones that occur over changes of harmony, such as accented dissonances and passing tones preceding a chord change, can be predicted with the help of metrical information and interval patterns associated with these types of dissonance figures. This explains the entries 11 and 13 on Table 1, which indicate a larger five-way redundancy between CT-M, SD-M, HC-LR, Int-LMR, and Met-LMR.

These observations help us interpret the ordering of features arrived at in the experiment. The importance of chord-tone status is reflected not only in the value of CT-M in the results of Experiment 2, but also in the large increases in sensitivity of the four features in Table 1. Therefore we may conclude that chord-tone status, as might be expected, plays a large role in the analytical process for these kinds of short excerpts.

5.2. *Removal of temporal features*

Trial 14, where the Lev1-LR feature is removed, is also prominent in Table 1. All of the large changes on this trial reflect interdependencies: entries 4 (Met-LMR), 6 (HC-LR), 16 (Int-LMR), and 21 (SD-M). These are all the features remaining in the model at trial 14. This result indicates that each of these basic metrical, harmonic, and melodic features operate differently at the local, note-to-note level than at higher levels. In other words, this is strong evidence that the rules of note-to-note analysis differ in systematic ways from higher-level analytical reasoning. For the harmonic and metrical features, this is readily understandable – a change of harmony at the note-to-note level is likely to represent a direct harmonic progression, and direct metrical relations, similarly, have a more specific meaning than indirect ones. However, the inclusion of melodic

Table 1. Changes of feature importance from one trial to the next greater than one standard deviation from the mean of 0.12% (less than −2.23% or greater than 2.47%).

No.	Feature	Trial	Feature removed	Edge accuracy	Difference from full	Change from previous trial
1	SD-M	13	CT-M	52.61	13.38	12.36
2	HC-LR	13	CT-M	52.74	13.25	7.77
3	SD-M	10	SD-LR	60.00	8.15	7.39
4	Met-LMR	14	Lev1-LR	60.00	0.51	– 5.99
5	CT-M	10	SD-LR	67.52	0.64	– 5.86
6	HC-LR	14	Lev1-LR	52.99	7.52	– 5.73
7	Int-LMR	10	SD-LR	59.36	8.79	5.73
8	Int-LMR	15	Met-LMR	45.73	14.27	5.22
9	MetN-LR	10	SD-LR	68.28	– 0.13	– 4.71
10	SD-M	11	MetN-LR	64.84	3.44	– 4.71
11	Int-LMR	13	CT-M	53.50	12.48	4.59
12	Int-LMR	12	Lev2-LR	58.85	7.90	– 4.08
13	Met-LMR	13	CT-M	59.49	6.50	4.20
14	Int-LMR	8	RN-LR	66.62	3.95	– 3.69
15	Met-LMR	11	MetN-LR	64.20	4.08	3.82
16	Int-LMR	14	Lev1-LR	51.46	9.04	– 3.44
17	Int-LMR	5	IntD-LR	65.35	7.64	3.44
18	SD-M	16	HC-LR	46.62	5.35	– 3.06
19	Met-LMR	10	SD-LR	67.90	0.25	– 3.06
20	Int-LMR	11	MetN-LR	56.31	11.97	3.18
21	SD-M	14	Lev1-LR	50.06	10.45	– 2.93
22	RN-LR	4	HC-M	72.36	0.76	– 2.68
23	SD-LR	9	RN-M	68.15	0.51	– 2.68
24	RN-LR	3	CT-LR	69.43	3.44	2.68
25	HC-LR	10	SD-LR	63.31	4.84	2.68
26	RN-M	8	RN-LR	68.66	1.91	– 2.42
27	SD-M	12	Lev2-LR	65.73	1.02	– 2.42

features also shows that we should expect different melodic shapes at slightly higher levels than at the note-to-note level.

The latter point may also be made about Lev2-LR, the distinction between within-measure and between-measure levels. The melodic features are also dependent on this feature as indicated in entries 12 (Int-LMR) and 27 (SD-M).

5.3. *Importance of Int-LMR*

The Int-LMR feature occurs frequently in Table 1. This data is perhaps most readily assessed by considering the overall trajectory of Int-LMR's feature importance, as shown in Figure 14. There is a general upward trajectory indicating that, as features are removed, Int-LMR compensates for many of them, especially melodic and metrical features. This trend is interrupted by three prominent dips. The first corresponds to the removal of two harmonic features, suggesting that the likelihood of certain interval patterns depends upon harmonic context (e.g. a change of harmony versus a stable harmony). The other two dips are for the temporal features just discussed.

5.4. *Interactions involving SD-LR*

As previously noted, the SD-LR feature is redundant with the combination of SD-M and Int-LMR, in the sense that the two middle features may be used to predict SD-LR, with the only exception being distinctions between chromatic and diatonic values. It is therefore unsurprising that two of the strongest interactions involve redundancies between these three features (entries

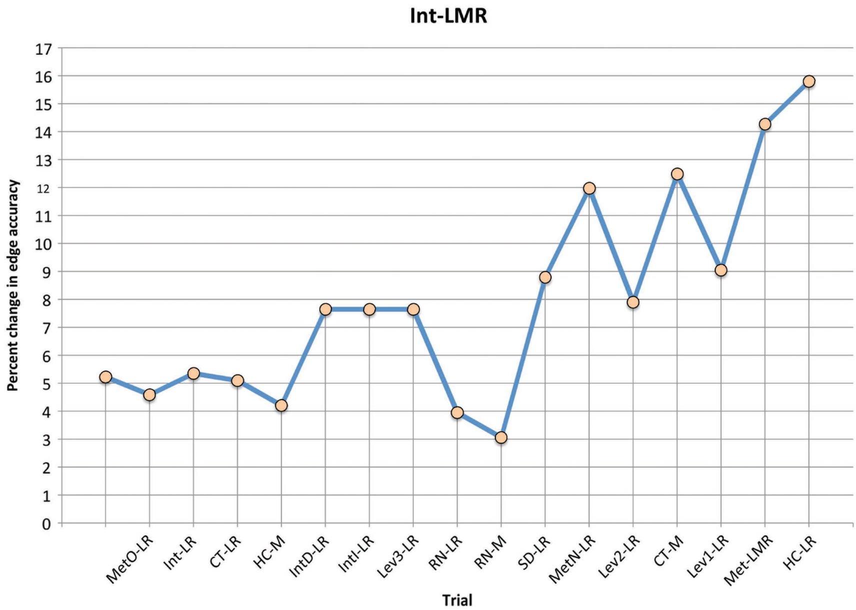

Figure 14. Changes in Int-LMR's feature importance through the trials. The labels on the horizontal axis show the feature removed prior to the given trial.

3 and 7 in Table 1). However, SD-LR also shows some prominent interdependencies indicated by negative shifts in entries 5 (CT-M), 9 (MetN-LR), and 19 (Met-LMR). These reflect the role of SD-LR as the only LR melodic feature remaining in the model beyond trial 6, which allows melodic considerations to appear on the left side of the conditional probability, as predictors. The results show that the use of melodic information to reliably predict features of the middle note is largely dependent upon complete harmonic (CT-M) and metrical (MetN-LR and Met-LMR) information.

6. Conclusions

- Melodic, harmonic, and metrical features are all significant considerations in Schenkerian analysis.
- As a whole, harmonic information is the most essential for making an accurate analysis.
- However, harmonic information breaks down into two separately important components. First, harmonic context is the most useful conditional feature for predicting characteristic patterns of all features. Second, identification of non-chord-tones is a significant aspect of the analytical process.
- Melodic information – scale-degree progression and intervallic pattern – is also crucial and actually becomes the most important analytical factor when harmonic features are divided up between harmonic context and chord tone status.
- Metrical information is the least important overall, but nonetheless does play an essential role.
- The rules of Schenkerian analysis vary from level to level. In particular, there is strong evidence that the note-to-note level follows substantially different rules than larger-scale levels. This is especially true of how melodic patterns work in Schenkerian analysis, which varies not only at the note-to-note level, but also between within-measure and across-measure levels.

Acknowledgments

The authors would like to thank the anonymous reviewers, José M. Iñesta (a Guest Editor of this special issue), and Thomas Fiore (the journal's co-Editor-in-Chief) for their helpful and insightful comments on earlier versions of this paper.

Disclosure statement

The authors have no conflict of interest.

References

Agawu, Kofi. 2009. *Music as Discourse: Semiotic Adventures in Romantic Music*. New York: Oxford University Press.

Breiman, Leo. 2001. "Random Forests." *Machine Learning* 45 (1): 5–32.

Burstein, L. Poundie. 2011. "Schenkerian Analysis and Occam's Razor." *Res Musica* 3 (3): 112–121.

Cadwallader, Allen, and David Gagné. 1998. *Analysis of Tonal Music: A Schenkerian Approach*. Oxford: Oxford University Press.

Forte, Allen, and Steven E. Gilbert. 1982a. *Instructor's Manual for "Introduction to Schenkerian Analysis*". New York: W. W. Norton.

Forte, Allen, and Steven E. Gilbert. 1982b. *Introduction to Schenkerian Analysis*. New York: W. W. Norton.

Frankel, R. E., S. J. Rosenschein, and S. W. Smoliar. 1978. "Schenker's Theory of Tonal Music – Its Explication Through Computational Processes." *International Journal of Man–Machine Studies* 10 (2): 121–138.

Gilbert, Édouard, and Darrell Conklin. 2007. "A Probabilistic Context-Free Grammar for Melodic Reduction." In *Proceedings of the International Workshop on Artificial Intelligence and Music, 20th International Joint Conference on Artificial Intelligence,* 83–94.

Hamanaka, Masatoshi, Keiji Hirata, and Satoshi Tojo. 2005. "ATTA: Automatic Time-Span Tree Analyzer Based on Extended GTTM." In *Proceedings of the Sixth International Conference on Music Information Retrieval,* 358–365.

Hamanaka, Masatoshi, Keiji Hirata, and Satoshi Tojo. 2006. "Implementing 'A Generative Theory of Tonal Music'." *Journal of New Music Research* 35 (4): 249–277.

Hamanaka, Masatoshi, Keiji Hirata, and Satoshi Tojo. 2007. "ATTA: Implementing GTTM on a Computer." In *Proceedings of the Eighth International Conference on Music Information Retrieval,* 285–286.

Hamanaka, Masatoshi, and Satoshi Tojo. 2009. "Interactive GTTM Analyzer." In *Proceedings of the 10th International Society for Music Information Retrieval Conference,* 291–296.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The Elements of Statistical Learning*. 2nd ed. New York: Springer-Verlag.

Jiménez, Víctor M., and Andrés Marzal. 2000. "Computation of the N Best Parse Trees for Weighted and Stochastic Context-Free Grammars." In *Advances in Pattern Recognition,* Volume 1876 of *Lecture Notes in Computer Science*, edited by Francesc J. Ferri, José M. Iñesta, Adnan Amin, and Pavel Pudil, 183–192. Berlin: Springer-Verlag.

Jurafsky, Daniel, and James H. Martin. 2009. *Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics*. 2nd ed. Upper Saddle River, NJ: Prentice-Hall.

Kanamori, Kouhei, Masatoshi Hamanaka, and Junichi Hoshino. 2014. "Method to Detect GTTM Local Grouping Boundaries Based on Clustering and Statistical Learning." In *Proceedings of the 11th Sound and Music Computing/40th International Computer Music Conference,* 1193–1197.

Kassler, Michael. 1975. *Proving Musical Theorems I: The Middleground of Heinrich Schenker's Theory of Tonality*. Technical Report 103, Basser Department of Computer Science, School of Physics, University of Sydney: Australia.

Kassler, Michael. 1987. "APL Applied in Music Theory." *APL Quote Quad* 18 (2): 209–214.

Kirlin, Phillip B. 2014a. "A Data Set for Computational Studies of Schenkerian Analysis." In *Proceedings of the 15th International Society for Music Information Retrieval Conference,* 213–218.

Kirlin, Phillip B. 2014b. "A Probabilistic Model of Hierarchical Music Analysis." Ph.D. thesis, University of Massachusetts Amherst.

Kirlin, Phillip B., and David D. Jensen. 2011. "Probabilistic Modeling of Hierarchical Music Analysis." In *Proceedings of the 12th International Society for Music Information Retrieval Conference,* 393–398.

Kirlin, Phillip B., and David D. Jensen. 2015. "Using Supervised Learning to Uncover Deep Musical Structure." In *Proceedings of the 29th AAAI Conference on Artificial Intelligence,* 1770–1776.
Lerdahl, Fred, and Ray Jackendoff. 1983. *A Generative Theory of Tonal Music*. Cambridge, MA: MIT Press.
Marsden, Alan. 2005. "Towards Schenkerian Analysis by Computer: A Reductional Matrix." In *Proceedings of the International Computer Music Conference,* 247–250.
Marsden, Alan. 2010. "Schenkerian Analysis by Computer: A Proof of Concept." *Journal of New Music Research* 39 (3): 269–289.
Marsden, Alan, and Geraint A. Wiggins. 2008. "Schenkerian Reduction as Search." In *Proceedings of the Fourth Conference on Interdisciplinary Musicology*, Thessaloniki, Greece.
Mavromatis, Panayotis, and Matthew Brown. 2004. "Parsing Context-Free Grammars for Music: A Computational Model of Schenkerian Analysis." In *Proceedings of the 8th International Conference on Music Perception & Cognition*, 414–415.
Miura, Yuji, Masatoshi Hamanaka, Keiji Hirata, and Satoshi Tojo. 2009. "Use of Decision Tree to Detect GTTM Group Boundaries." In *Proceedings of the International Computer Music Conference,* 125–128.
Pankhurst, Tom. 2008. *SchenkerGUIDE: A Brief Handbook and Website for Schenkerian Analysis*. New York: Routledge.
Provost, Foster, and Pedro Domingos. 2003. "Tree Induction for Probability-Based Ranking." *Machine Learning* 52 (3): 199–215.
Rothstein, William. 1990. "The Americanization of Schenker Pedagogy?" *Journal of Music Theory Pedagogy* 4 (2): 295–300.
Schachter, Carl. 1990. "Either/Or." In *Schenker Studies*, Volume 1, edited by Hedi Siegel, 165–179. Cambridge, UK: Cambridge University Press.
Yust, Jason. 2006. "Formal Models of Prolongation." Ph.D. thesis, University of Washington, Seattle.
Yust, Jason. 2009. "The Geometry of Melodic, Harmonic, and Metrical Hierarchy." In *Proceedings of the International Conference on Mathematics and Computation in Music*, 180–192.
Yust, Jason. 2015. "Voice-Leading Transformation and Generative Theories of Tonal Structure." *Music Theory Online* 21 (4). http://www.mtosmt.org/issues/mto.15.21.4/mto.15.21.4.yust.html.