DIGITAL ACCESS TO SCHOLARSHIP AT HARVARD

SCHUBOT: Machine Learning Tools for the Automated Analysis of Schubert's Lieder

(Article begins on next page)

SCHUBOT: Machine Learning Tools for the

Automated Analysis of Schubert's Lieder

by Dylan J. Nagler '14

submitted in partial fulfillment of the requirements for an AB degree

with honors in Computer Science and Music

Departments of Computer Science and Music

Harvard College

March 3, 2014

ABSTRACT

This paper compares various methods for automated musical analysis, applying machine learning techniques to gain insight about the Lieder (art songs) of composer Franz Schubert (1797-1828). Known as a rule-breaking, individualistic, and adventurous composer, Schubert produced hundreds of emotionally-charged songs that have challenged music theorists to this day. The algorithms presented in this paper analyze the harmonies, melodies, and texts of these songs.

This paper begins with an exploration of the relevant music theory and machine learning algorithms (Chapter 1), alongside a general discussion of the place Schubert holds within the world of music theory. The focus is then turned to automated harmonic analysis and hierarchical decomposition of MusicXML data, presenting new algorithms for phrase-based analysis in the context of past research (Chapter 2). Melodic analysis is then discussed (Chapter 3), using unsupervised clustering methods as a complement to harmonic analyses. This paper then seeks to analyze the texts Schubert chose for his songs in the context of the songs' relevant musical features (Chapter 4), combining natural language processing with feature extraction to pinpoint trends in Schubert's career.

Table of Contents

CHAPTER 1

Introduction:

Tools for Understanding Schubert's Harmonic Language

This chapter is included to provide a cursory understanding of the relevant music theory and computer science utilized throughout this paper. My aim in this chapter is to eschew technical detail for the sake of accessibility; readers familiar with both fields may wish to forego the first and final two sections of this chapter.

A Brief Explanation of Tonal Harmony

Western music is built on an alphabet of 12 *pitch classes*: {C, C#/D♭, D, D#/E♭, E, F, F#/G♭, G, G#/A♭, A, A#/B♭, B}. In computational musicology, these pitch classes are often bijected with the integers from 0 to 11. These pitch classes are presented in order of ascending *pitch*, or fundamental frequency, with an *octave*, or doubling of frequency, between adjacent members of the same pitch class (eg. C4 and C5, where the number denotes the octave). In the *equal temperament* tuning system, the difference between two adjacent pitches (eg. C4 and C#4) is normalized to one *half-step* or *semitone*, with a *whole step/tone* between pitches such as C4 and D4. In equal temperament, pitches such as C#4 and D♭4 are called *enharmonic*, and are considered to be the same pitch.

Tonal music is defined by the prevalence of one pitch class as a *tonic* around which the melody and harmony are built. The two most fundamental structures in tonal music are the *scale* and the *chord*, which both serve as collections of pitch classes. Given a tonic pitch class, two basic scales can be formed as ordered collections of pitch classes: a major scale, following the step pattern of whole-whole-half-whole-whole-whole-half, and a natural minor scale, following the pattern of whole-half-whole-whole-half-whole-whole. For example, the C major scale consists of {C, D, E, F, G, A, B, C}, and the A natural minor scale consists of {A, B, C, D, E, F, G, A}. While the natural minor scale is generally used for modeling theoretical principles, two variants of the minor scale are popular in composition. The harmonic minor scale contains a raised seventh scale degree, or *leading tone*,

and the melodic minor scale contains raised sixth and seventh scale degrees when ascending but plays as a natural minor scale when descending. *Intervals* are defined by the distances in half-steps between the root of a scale and its other pitches; for example, a major third is the distance between the root and third degree of a major scale (4 half-steps) and a minor sixth is the distance between the root and sixth degree of a minor scale (8 half-steps). Intervals are said to be *inversions* if their half-step distances sum to 0 mod 12. The choice of scale given the tonic is known as *modality*; the choice of tonic and scale together is known as the *key*.

For the purpose of this paper, a chord is defined as any collection of three or more pitch classes that sound simultaneously or within a small span of time. Given a scale, simple chords known as *triads* may be formed by taking a *root* note from the scale and adding to the set the notes 2 and 4 scale degrees above it. For example, a triad in the key of C major may consist of the pitches {C, E, G}. Within a tonal context, triads may be assigned Roman numerals based on their root notes. For example, a C major triad {C, E, G} is assigned the numeral I within the key of C major, and an A minor triad {A, C, E} is assigned the numeral vi, with lowercase numerals indicating minor triads.

Perhaps the most fundamental aspect of tonal *harmony* (or the use of chords in progressions) is that triads serve *functions*. In 1722, composer Jean-Philippe Rameau published his revolutionary "Treatise on Harmony", presenting the triad as the most primal element of music (Burkholder 2010:430). Rameau argued that within a tonal context, the most important progression of chords was from V to I or i, with V serving a *dominant* function that resolves to the tonic. Rameau defined musical phrases (the equivalent of sentences in natural languages) as a motion from the tonic (I), to a predominant (ii or IV), to a dominant (V), followed by a *cadence* back to I. Note that Rameau's theory was as prescriptive as it was descriptive of the work of his peers; it did a great deal to codify the style of his contemporaries and successors.

More recent theories of tonal music have served descriptive functions, noting the habits of composers in the "common practice" period of the 18th and 19th centuries as defined in Walter Piston's 1944 treatise "Harmony". Piston argued that theory must follow practice, and presented tonal harmony in terms of Roman numerals

and their embellishments. Like Rameau, Piston viewed the cadence as the heart of tonal music; he observed that the endings of phrases marked places of musical stability while establishing the tonality and rendering the formal structure of the work coherent (Piston 1944:125). It is worth noting here that any theory of functional harmony is necessarily hierarchical, with some chords and some progressions seen as more important than others. This sense of hierarchy is fundamental to the understanding of large-scale musical forms, as later theories suggest.

Modern music theorist Dmitri Tymoczko has made efforts to broaden the perception of tonal practices, presenting a controversial argument for the consideration of an "extended common practice" period from the 11th century to the present day (Tymoczko 2011:195). His "Geometry of Music" recasts functional harmony as movements within spaces of varying dimensions. He generalizes the concept of *voice leading*, the parsimonious movement between triads such as {C, E, G} → {C, E, A}, to movement between scales, which he uses as "musical rulers" (ibid:116). Generalizing the idea that the distance between chords can be equated to the number of single-note voice leadings needed to move between them, he argues that distances between keys are equivalent to voice-leading distances between scales (ibid:247). On the topic of hierarchical structures, he acknowledges that tonal music is hierarchically self-similar in that harmonic consistency and efficient voice leading exist both at the level of the chord and of the scale.

Hierarchical Musical Analysis

Beyond the level of local movement between chords, tonal music is marked by its *modulations*, or movements between keys. Modulation is at the heart of music written in the common practice period; as Piston puts it, "composers appear to have been in consistent agreement that to remain in one key throughout a piece of any length is aesthetically undesirable" (ibid:77). Modulations most often occur between "closely related" keys: for example, a movement from C major to G major entails a change of one pitch class (F → F#) in the respective keys' scales. In terms of chords, modulation involves changing from the set of triads belonging to one key to the set belonging to another. Smooth modulations often occur through the use of a *pivot chord*, a triad that serves an important function in both keys. In the C

major to G major example, the A minor triad serves as both vi in C and ii (functioning as a predominant) in G. In most formally tonal music, pieces begin and end in the same key, often following an overall key progression of I-V-I. Author Douglas Hofstadter compared such constructions to a recursive stack in a computer program, with new key areas "pushed" onto the stack and "popped" with each modulation, beginning and ending in the tonic (Hofstadter 1979:129). While admitting this to be an exaggeration, Hofstadter goes on to explain that this whimsical comparison exemplifies the perception of tension and resolution at both local and global levels within a tonal work.

Hofstadter's analogy is perfectly in line with the writings of influential 20th-century music theorist Heinrich Schenker, who believed that tonal music has a simple fundamental background structure known as the *Ursatz*. Schenker's theory largely disregarded the rhythmic construction of a piece (Schachter 1999:17), favoring the analysis of counterpoint (note-against-note intervallic motion) within the context of tonal harmony. Schenker's analyses were reductive and hierarchical, repeatedly simplifying the harmonic and contrapuntal motions in a work to higher-level abstractions. Schenker's hierarchical analyses relied heavily on the assumption that a tonal work will begin and end in the same key (if only at an abstract background level); he viewed temporary key changes as "higher unities in the foreground" (ibid:134). Schachter argues that performing reductive analysis is by no means a deterministic process; ambiguities do occur in tonal music, though Schachter insists that they are far less quarrelsome than ambiguities in natural language (ibid:121). According to Schachter, at each ambiguity encountered in the parsing and reduction of a piece, a choice must be made as to whether a change is local or prolongational. In particular, he argues that Schenkerian analysis prefers the placement of boundaries between prolongational spans such that the boundary coincides with a point of large-scale formal articulation (ibid:126). Much like the concept of a pivot chord, Schenkerian "pivots" can exist both in the foreground (more local structure) or the background (large-scale structure).

In his 1973 work "Explaining Music", critic Leonard B. Meyer theorized that hierarchical structures in music can arise only if the piece can be parsed in discrete segments at varying levels of such a hierarchy (Meyer 1973:81). Meyer noted that

closure is necessary for this sense of discreteness, and that cadences bring about both harmonic and rhythmic closure (ibid:85). As Meyer sees it, "the structure of a composition is something which we infer from the hierarchy of closures which it presents. A composition continues – is mobile and on-going – partly because of the tendency of parameters to act independently of one another, to be noncongruent" (ibid:89). Meyer's analyses, unlike those of Schenker, focus largely on the implications of tonal melody and harmony at the foreground (ibid:110). Without proposing any formal system of analysis, however, Meyer acknowledges that the analysis of hierarchical patterns involves the difficult problem of establishing a reasonably objective grounds for distinguishing structural events from ornamental events (ibid:121).

At the core of many modern perceptions of hierarchy and reduction in tonal music is the 1983 treatise "A Generative Theory of Tonal Music" by Fred Lerdahl and Ray Jackendoff. Known henceforth in this paper as *GTTM*, this theory sought to outline hierarchies composed of discrete non-overlapping elements or regions such that one may subsume or contain another (Lerdahl, Jackendoff 1983). These *hierarchical trees* expanded on Schenkerian prolongational reductions to consider grouping, metrical, and time-span reductions as well. By establishing preference, transformational, and well-formedness rules, the two authors provided effective segmentations of tonal works into their constituent parts at various levels. Lerdahl's subsequent work focused on refining this theory, providing concrete distance metrics between chords at different levels (Lerdahl 1988). Expanding on the event hierarchies discussed in *GTTM*, Lerdahl discussed the tonal hierarchies of his "tonal pitch space", arguing that his metrics are rooted in human perception (ibid:337). Jackendoff's 1991 paper "Musical Parsing and Musical Affect" theorizes the real-time parsing of a piece by a listener in terms similar to the *GTTM* – perception is modeled as a sequence of selections at points of indeterminacy as part of a larger hierarchical structure. The prospects of formalizing theories such as *GTTM* into computer applications for automated analyses are discussed at length in Chapter 2 of this paper.

## Descriptive Theories of Schubert's Harmony and the Issue of Modulation

The works of Austrian composer Franz Schubert have played a unique role in the world of tonal harmonic analysis since the time of their composition. Schubert was at the forefront of the 19th-century Romantic movement, marked by an unprecedented interest in individuality and emotion in the arts. Schubert was renowned for his Lieder, or art songs for voice and piano; his settings of the poems of his contemporaries contained "beautiful melodies that perfectly capture a poem's character, mood, and situation" (Burkholder 2010:609). In terms of tonal harmony, the Romantic movement saw the rise of increased *chromaticism*, the use of new harmonic progressions based on parsimonious voice-leading that often broke the rules of Classical tonal harmony. Described as a "somnambulant" or "clairvoyant" composer by many of his contemporary critics (Clark 2011:6, 53), his works were perceived as somewhat out of touch with music theory and often left unquestioned. Schubert's detractors took issue with the frequent and often jarring modulations he employed in his songs, though it has been argued that it was his method of modulating, and not his choice of keys, that upset his critics (ibid:57).

An interesting problem posed to analysts of Schubert is that many of his works begin and end in different keys. Attempts have been made to reconcile this with Schenker's *Ursatz*; a modulation that is never "popped", to use Hofstadter's terminology, can be equated to an "interrupted" Schenkerian structure (ibid:73), or to a combination of two *Ursätze*, each present at a different level (ibid:109). On the topic of Schubert's songs, Schachter makes no mention of "unpopped" modulations, but presents Schenkerian analyses that outline the songs' texts and their developments of *motives*, or salient melodic fragments (Schachter 1999:209). Modern theorist Richard Cohn notes Schachter's reliance on motives in resolving analytical ambiguities, and goes on to argue that the consideration of motivic relations is unavoidable in Schenkerian analysis (Cohn 1992:156). Cohn claims that motivic cause-and-effect relationships remain autonomous in Schenkerian analysis, cutting across the prolongational hierarchy without moving towards or away from the *Ursatz* (ibid:166). In line with this argument, many successful Schenkerian analyses have been made of

Schubert's songs, often relying on motives as a defining feature. For example, analyst Walter Everett argued that the songs of Schubert's cycle *Die Winterreise* are connected by a single three-note motive that expresses the singer's grief (Everett 1990:157). Chapter 3 of this paper focuses on the integration of motivic analysis in the structural analysis of a tonal work.

Perhaps a more useful theoretical framework for Schubert's songs is that of neo-Riemannian analysis, which presents a new metric for proximity between triads: a torus-shaped *Tonnetz* in which pitches are connected by thirds and fifths.
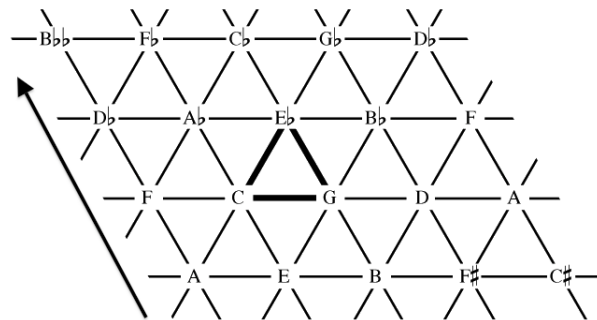
(Image courtesy of Wikimedia Commons)

Here, triads are outlined by triangles, with surprisingly small distances appearing between seemingly unrelated triads. Richard Cohn has proposed that Schubert's use of tonality is best understood when chords are linked by the common pitch classes they contain (Cohn 1999:214). Along these lines, Cohn divides the 24 major and minor triads into four "hemispheres" of cyclically-related sonorities (ibid:216). Within each hemisphere, chords are connected by a single-semitone voice leading; for example, the Northern hemisphere contains the cycle {E major → E minor → C major → C minor → A♭ major → G# minor}.
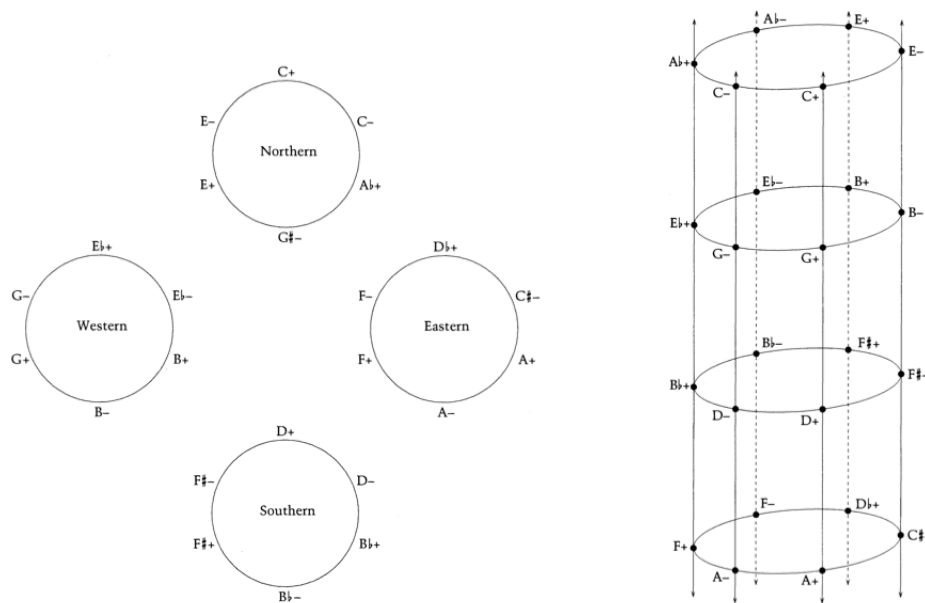
Cohn goes on to point out that the four hemispheres are each related by fifth to two others, creating a new torus-shaped structure (above right) that closely resembles the *Tonnetz*. Note how the vertical axes in this figure contain the same fifth relations as the horizontal axes in the *Tonnetz*. Similarly, the roots of the triads on each circle in this figure correspond to a line on the *Tonnetz* in the direction indicated by the arrow. Tymoczko's analysis of Schubert's harmony falls in agreement with this, noting Schubert's use of major-third root progressions (Tymoczko 2011:280) and the underlying principle that efficient voice leading can yield *substitutions* that do little to disrupt a piece's harmonic fabric (ibid:283). Neo-Riemannian transformations make for convenient labels, as although they are often used for surface-level analyses, they are easily adapted to operate at a higher hierarchical level. In addition, they rely only on the context of their immediate harmonic predecessors, and not a global tonic, as Roman numerals do (Clark 2011:71-72). The efficacy of augmenting traditional rules for harmonic motion with neo-Riemannian transformations is discussed in Chapter 2.

A Primer on Machine Learning Algorithms

Switching gears from the music theoretical discussion that dominated the first half of this chapter, the remainder of the chapter seeks to establish a fundamental understanding of the various machine learning algorithms that will be used in subsequent chapters. These descriptions are intentionally non-technical, so as to be accessible without fully revealing the mechanics of the underlying formulae.

As a subfield of artificial intelligence (AI), machine learning is concerned with the development of algorithms that can *learn* hypotheses from data sets. Many machine learning approaches can be classified into two categories: supervised and unsupervised. Supervised learning is concerned with the *classification* problem: given a set of data points that each belong to one of any number of *classes*, how can one generalize a hypothesis that will properly classify unseen data points? When using supervised learning algorithms, the data set is partitioned into *testing* and *training* sets so as to avoid the formation of ad hoc hypotheses that do not generalize.

The use of *decision trees* for prediction is a simple supervised learning method that is utilized in Chapter 2. The idea behind decision trees is that classification comes after a series of considerations of individual data attributes. Decision trees are constructed ("learned") through algorithms such as ID3 that focus on maximizing *information gain* at each node in the tree. Decision trees are particularly useful for decisions made on binary (yes/no) variables, but can easily suffer from *overfitting*, the phenomenon of formulating overly complex hypotheses so as to best fit the training data.

Another simple classified learning method is the k-nearest neighbor algorithm: given a constant k and an unclassified data point, consider the k data points closest to it (using some pre-defined distance metric) and classify it by simple majority vote. The k-NN algorithm is used for the sake of comparison in Chapter 2.

A more robust classified learning method is the use of (*artificial*) *neural networks*, based loosely on the functioning of human brains at a cellular level. The idea behind neural network learning is to consider a linear combination of simple hypotheses; the input "neurons", known as *perceptrons*, each create a simple (often binary) decision based on an attribute, and the result of this decision is propagated

forward in the network. The network then uses a backwards algorithm, calculating the error in the perceptrons' collective decision and reassigning weights to each individual decision so as to decrease error.

Whereas supervised learning focuses on tuning parameters in complex equations so as to improve accuracy and generalization, unsupervised learning seeks to establish categories or elucidate hidden structures for unclassified data. This is often done through clustering, or grouping related data points based on their similarity. Hierarchical agglomerative clustering (HAC) begins with each data point as its own singleton cluster and iteratively merges the two clusters that are closest according to a given distance metric. HAC may be stopped after a specified number of clusters is reached, or may continue so as to form a complete hierarchy of clusters. HAC clustering is utilized in Chapter 3 of this paper.

Another clustering algorithm is k-means clustering, which begins by assigning k randomly chosen data points as *prototypes* for clusters. At each iteration of the algorithm, each data point is clustered with the nearest prototype, and the mean of each cluster is recalculated. k-means clustering runs until convergence (namely, until there exists an iteration in which no data point is reassigned to a new cluster), and can thus be computationally very complex.

Outside of the domain of classification and clustering, another useful machine learning model is the hidden Markov model (HMM). HMMs are useful for data points that transition between states (eg: chords within tonal harmony). In a simple Markov model, transitions between observed states are explained by the probability of their occurrence; in such a model, only the current state is considered when making a transition, though more complex variations can be used. The idea behind a *hidden* Markov model is that the observations are separated from the (hidden) stochastic states, with each state holding its own distribution of possible observation probabilities. These probabilities are learned with a backwards-forwards expectation-maximization algorithm. See Chapter 2 for a discussion of HMMs in harmonic analysis.

At the intersection of machine learning and natural language processing (NLP) is the Latent Dirichlet Allocation algorithm (LDA) (Blei 2003). LDA is a model that generates "topics" from texts by treating them as "bags of words", or unordered

sets of words with their frequencies. LDA seeks to discover latent variables in the distribution of words amongst the set of texts, viewing each text as a combination of topics. As such, each word in a text is "generated" by one or more of the underlying topics. LDA is useful for unsupervised sentiment analysis; in Chapter 4, the algorithm is used to analyze the emotions of the texts Schubert set in his songs.

An Overview of Context-Free Grammars

Given a language composed of strings of symbols ("terminals"), formal grammars serve the purpose of outlining the means by which strings in the language may be produced. The transitions between the hidden states of an HMM that yield observances form a simple formal grammar. Context-free grammars (CFGs) are a more restricted formal grammar, in which *all* production rules are of the form $A \to b$, where A is a single non-terminal symbol and b is a string of terminal and non-terminal symbols that may be empty. A non-terminal is not a symbol in the language's alphabet, but may be thought of as a predecessor to a number of symbols. CFGs require finite sets of non-terminals, terminals, and production rules, as well as a single start symbol.

A CFG is said to be in Chomsky Normal Form (CNF) if all production rules are of the form $\{A \to BC, A \to b, A \to \varepsilon\}$, where A, B, C are non-terminals, b is a single terminal, and ε denotes the empty string. All CFGs can be converted into CNF; it allows for ease of parsing, as with the CYK algorithm (see Chapter 2). A probabilistic/stochastic context-free grammar (PCFG) is a CFG in which each production rule is associated with a probability. The efficacy of CFGS and PCFGs in the analysis of tonal music is discussed in Chapter 2; they are hierarchical in nature, but it can be argued that they are too formal for the purposes of musical analysis.

CHAPTER 2

Phrase-Based Harmonic Analysis

An Overview of Supervised Learning in Automated Harmonic Analysis

While much of the original research presented in this chapter will focus on the applications of unsupervised learning in automated harmonic analysis, we will begin with a discussion of the efficacy of supervised learning. While most musical analysis performed by hand is unsupervised by nature, supervised methods that can generalize their results to unseen data can prove to be useful.

Neural networks have been shown to be an effective aid in automated harmonic analysis. Hörnel and Menzel (1998) discuss how neural networks can be applied to the problem of learning a composer's style. Here, the output classifications of a neural network are used as approximated posterior distributions for harmonic progressions. This neural network approach is applied to the automated harmonization of a chorale melody in the style of composer J. S. Bach. As will be discussed later in this chapter, Bach's chorales make for an excellent case study in supervised learning, as their harmonies are essentially Markovian (following directly from the previous element in a sequence) and their segmentation points (phrase boundaries) are given in the score.

In the same vein, Heinrich Taube's 1999 paper "Automated Tonal Analysis: Toward the Implementation of a Music Theory Workbench" uses the Bach chorales as a corpus for harmonic learning. Taube proposes a four-pass analysis of the chorales, analyzing sonorities, then dealing with non-harmonic tones before performing an automated harmonic analysis based on local root motion and Roman numeral labeling. This kind of analysis is ideal for phrase-based compositions such as the Bach chorales, for which each phrase is largely self-contained and the tonal hierarchy is relatively flat. However, Palmer (1997:123) indicates that neural networks can prove useful for hierarchical analysis within the field of performance analysis.

Returning to the topic of Schenkerian analysis of hierarchical structure, Kirlin and Jensen (2011) presents a supervised means of representing and predicting re-

ductions of tonal works. Their paper begins by constructing *maximal outerplanar graphs* (MOPs), for pre-existing reductive analyses of melodies. MOPs can be seen as compact representations of hierarchical trees. Using these MOPs and OPCs (outerplanar graph collections), they create a joint probabilistic distribution of graph nodes that can be used to predict the presence of triangles in such analysis graphs. This field of research is still developing; the available data set of computerized Schenkerian analyses is very small, and their model can only account for isolated melodies without any sort of harmonic context.

Computational Musicology with music21

The new machine learning methods presented in this paper rely on the music21 Python library, an open-source toolkit for computational musicology developed by MIT Prof. Michael Scott Cuthbert. music21 works best with MusicXML input; its parser converts the metadata into a hierarchical Python object known as a stream which includes all notes and metadata of the score stored in parts, measures, groupings, etc.

The raw data for this paper comes courtesy of Prof. Cuthbert, and is taken from Yale University's MIDI archive. Given that music21 does not (at present) deal with MIDI input directly, I performed some pre-processing, using a plug-in for the software Sibelius 6 to convert the MIDI files to Sibelius scores, which were then converted to MusicXML documents. Some manual "cleaning" of the data was performed at this stage, correcting errors in the .midi to .sib conversion and ensuring that each Sibelius score passed a basic comparison to a notated score for the piece in IMSLP's Petrucci library.

Separation of Melody and Harmony: The Chord as an Atomic Unit

My automated harmonic analyses of Schubert's Lieder begin with the consideration of harmony in isolation. Each score in the corpus consists of two parts: a melody to be sung and an accompaniment to be played on the piano. Schubert's piano writing is often triadic, clearly marking each chord in a song with a simultaneous sounding of its constituent pitch classes. Note that for songs which tend to arpeggiate the chords, or sound their pitches in a non-simultaneous pattern, I have

taken the liberty of adding time-aligned "block" chords to the score as part of my pre-processing.

Given the history of chord-based theories of tonal harmony, and given the functionality of music21's chord object, I will use chords as the "atoms" of my harmonic analysis, seeking to explain the progressions of chords in Schubert's songs both at a local (chord-to-chord) and a global (song reduction) level. Viewing songs as sequences of chords (and not individual notes) is a convenient abstraction for the analyses performed in this chapter, as it separates the "noisy" texture of a song from its functional core.

Case Study: Predicting Cadences in Bach's Chorales with Decision Trees

As a precursor to my study of Schubert's work, I developed an algorithm for analyzing the chorales of J. S. Bach, a composer renowned for his beautiful yet highly rule-based compositions. As part of my research partnership with Prof. Cuthbert at the Radcliffe Institute, I developed a script that used various music21 functionality to predict the placement of phrase boundaries in Bach's chorales. In these chorales, the four voices move relatively *homophonically*, supporting each note in the melody with three simultaneous notes in the lower voices that spell out a triad or four-note chord. To find phrase segmentations, I relied on the fact that Bach marked the cadence at the end of each phrase with a fermata (a marking indicating a note held longer than its usual value). Given exact knowledge of which chords did or did not have fermatas, this became a relatively straight-forward supervised learning problem.

Using music21's built-in Bach chorale corpus and its iterator functionality, I logged the following 13 attributes for each note in the melody voice of each chorale:

1. Length, in quarter notes, from the beginning of the score

2. Quarter-note offset divided by total length of score

3. Beat strength (a music21 feature that assigns a value to a beat given its meter)

4. Cadence signature (see below)

5. Stepwise melodic change over the last two melody notes (in whole steps)

6. Stepwise melodic change over the last three melody notes

7. Length, in quarter notes, of the note itself

8. Number of sharps or flats in the key signature (a one-to-one correspondence with each scale)

9. Key signature mode (a binary variable – major or minor)

10. Time signature numerator

11. Time signature denominator

12. Total number of fermatas in the piece

13. Fermata presence (a binary variable – true or false)

Cadence signature, the most important attribute, is an integer between 0 and 3 that examines the 3-gram of the Roman numeral functions of the three most recent chords. Note that this Roman numeral analysis is *localized* to the present chord, making the naïve assumption that the present chord represents the current key that the piece is in. A value of 1 indicates an *authentic* cadence to a local tonic, with a Roman numeral signature of (*-V-I), where * denotes the possibility of any tonic, predominant, or dominant-functioning chord. A value of 2 indicates a *half* cadence, or movement to the dominant, with a signature of (*-IV-I), (*-iv-I), or (*-♭VII-I); again, assuming that the current chord is the tonic, a global dominant would see a subdominant as a ♭VII chord. A value of 3 indicates a deceptive cadence, of the form (♭VI-♭VII-i), and a value of 0 encapsulates all remaining 3-grams as inconclusive.

After calculating the 13 attributes of each chord, the script writes the data to comma-separated value (CSV) files, with each alternating chorale added to either the training set or the test set. These CSVs are then fed as input to a second script which utilizes Python's Orange and OrngTree libraries for classified learning. The script uses a simple majority learner (always favoring the more prevalent classification), which in this case is the classification of no fermata, along with a decision tree learner and a *k*-Nearest Neighbor learner. The results are as follows:

TRAINING SET

	Accuracy	Sensitivity	Specificity
Majority	0.878	0.000	0.100
Decision Tree	0.978	0.824	0.100
k-Nearest Neighbor	0.914	0.338	0.994

TESTING SET

	Accuracy	Sensitivity	Specificity
Majority	0.896	0.000	0.100
Decision Tree	0.881	0.487	0.927
k-Nearest Neighbor	0.916	0.397	0.976

These results show a drastic under-classification of fermatas. These decision trees may have suffered from overfitting, as the cadence signature attribute is extremely telling in the context of Bach's chorales.

Extending Decision Trees to Schubert

Despite the low sensitivity values in my Bach analysis, the accuracy of these results seemed somewhat promising. I decided to repurpose my scripts for Schubert's songs so as to serve as a performance baseline for other phrase-boundary predictions. I was curious to see the effects of decision trees, which take each data point (chord) out of its temporal context in the song's harmonic progression. I was wary of such naïve decision-making; whereas Bach's chorales have relatively flat hierarchies, with most chords serving an important harmonic function, Schubert's songs are fundamentally hierarchical in their tonality, as discussed in the previous chapter. As such, a local I-V-I motion may *not* count as a phrase ending, but rather as a *tonic prolongation* within a longer phrase. In addition, Schubert's phrases do not always end in a traditional cadential signature, as Bach's tend to. Given that Schubert's melodies do not generally move homophonically with their accompaniment (with melody notes often occupying several chords, and vice versa), the simple one-to-one melodic attributes cannot be calculated for Schubert's chords. As such, my 8 chord attributes were:

1. Key

2. Length, in quarter notes, from the beginning of the score

3. Chord Quality (major, minor, or otherwise)

4. Pitched common name (eg: C major triad)

5. Ordered pitch classes (a sorted list, eg: [0, 4, 7] for a C major triad)

6. Length, in quarter notes, of the chord itself

7. Cadence signature

8. "Fermata" presence

I placed fermatas in the scores myself using Sibelius 6, making a judgement of where each phrase in the song begins and ends. I used as my data set the 24 songs of Schubert's 1823 song cycle *Die Winterreise*, with 18 songs in the training set and 6 songs in the testing set. The results are as follows:

TRAINING SET

	Accuracy	Sensitivity	Specificity
Majority	0.905	0.000	0.100
Decision Tree	0.978	0.773	0.100
k-Nearest Neighbor	0.914	0.126	0.997

TESTING SET

	Accuracy	Sensitivity	Specificity
Majority	0.928	0.000	0.100
Decision Tree	0.958	0.690	0.978
k-Nearest Neighbor	0.928	0.172	0.987

The results were much better than I expected, but will not suffice for our purposes. Given the relative scarcity of phrase boundaries within a piece, it is not surprising that decision trees and k-nearest neighbor methods will both often assume that any given chord is not at a phrase boundary. Of course, the choice of supervised learning here is somewhat counter to our goal of automated analysis; it is included in this study as a baseline.

Markov Models for Tonal Harmony

We now shift our focus to the *unsupervised* learning of tonal harmonic structure. We begin by examining the use of stochastic models for chord progressions, whether HMMs or not. Beginning with Rameau's theory of functional harmony, transition rules have been established for movement between harmonies. Piston (1944:17) presents a "Table of Usual Root Progressions", each of the form "x followed by y, or sometimes z, less often w." This is, in effect, a simple Markov model in which the transition probabilities between states (harmonies) are defined by vague boundaries.

Ponsford (1999) explores the use of HMMs in the analysis of a corpus of *sarabandes*, 17th-century European dances. In their paper, overlapping 3-grams and 4-grams of chords are considered to be atomic units, where transitional probabilities are marked between these local progressions. As part of their pre-processing, they tokenized chords by sampling at each quarter note and elided adjacent chords if they were equivalent (a procedure that I followed in my decision tree learning of Schubert above). The goal of their experiment was to learn a basic grammar for harmonic transitions, which can be modeled by a finite-state transaction network graph. Ponsford *et al.* raise an interesting point of declaring PCFGs "unnecessarily powerful" for their purposes due to the lack of structural recursion in their corpus. Namely, in a CFG, rules of the form $\{A \to B, B \to AA\}$ are allowed, akin to the segmentation of phrases into smaller phrases in more hierarchical works. It may be asserted that Markov models do not lend themselves well to hierarchical analyses, as the Markov assumption that the past and future are conditionally independent given the present goes against the previously discussed models of tonal musical perception and structure. The efficacy of (P)CFGs in this sort of analysis is discussed in the following sections of this chapter.

Raphael and Stoddard (2004) present another application of HMMs for harmonic analysis in the field of signal processing. Given MIDI inputs, their algorithm provides a chord-by-chord segmentation, classifying each note with a key area and a Roman numeral based on a corpus of training inputs. While this thesis is not concerned with the issue of signal processing (as it takes MIDI to MusicXML in pre-processing), Raphael and Stoddard's paper is of note due to its use of a "bag of

words" model for individual notes; this type of model is discussed further in Chapter 4.

In a related vein, Wei Chai's 2005 thesis on "Automated Analysis of Musical Structure" takes an HMM signal processing approach to the automated structural analysis of pop music. Much like Raphael, Chai uses HMMs to detect scale, key, and chord progression given an input of recorded audio. Chai's method of hierarchical analysis uses a dynamic programming algorithm to compute sampling distances between various portions of a song; this is slightly less feasible in my corpus of notated scores, for which distance metrics are harder to formulate than simple differences between signals. Regardless, Chai's analysis provides a useful model for hierarchical decomposition: with knowledge of the number of major formal sections in a song, HAC and k-means clustering may be employed to group similar sections of a work together. Chai goes on to outline top-down and bottom-up algorithms for analyzing repetitive song structure. The final portion of Chai's thesis is related to my work in Chapter 3; given a segmentation of a piece, heuristics for finding "hooks" or motives can be established. While Chai's end goal is very similar to mine, our works may be differentiated by their inputs. The analysis of notated music, as opposed to the signal processing of performances, may ultimately prove more useful and generalizable in shedding light on the composer's intentions.

Víctor Gabriel Adán's thesis on "Hierarchical Music Structure Analysis, Modeling, and Resynthesis: A Dynamical Systems and Signal Processing Approach", published in the same year, employs techniques similar to Chai's with the end goal of generating curves representing the algorithm's inputs that can be hybridized and otherwise altered. Adán uses a stochastic model that represents the music's trajectory with respect to latent factors. Adán's hierarchical signal decomposition focuses on breaking the piece into abstract component parts that do not necessarily shed light on its formal structure.

Descriptive Grammars for Tonal Harmony

The formalization of grammars for tonal harmony can be traced back to the linguist Terry Winograd, whose 1968 paper "Linguistics and the Analysis of Tonal Harmony" discussed tonal music's position within the field of natural language processing (NLP). Winograd argues that while music is, technically speaking, a natural language that evolved independently of linguistic theory, we may still take a formal approach to its analysis. Winograd argues against the simple (Markovian, chord-by-chord) syntax of traditional tonal analysis, following the example of Noam Chomsky's generative grammars for formal languages. Winograd's grammar is somewhat ambiguous, allowing for the creations of multiple parsings of a work that are compared with a plausibility metric. Winograd shows successful results for inputs by both Bach and Schubert, leaving room for further experimentation on my part.

Recently, a simple context-free grammar for tonal harmony has been presented (Magalhaes, de Haas 2011); by limiting its scope to non-modulatory analysis, their HARMTRACE grammar can be summarized in just a handful of generative rules. Koops (2012) shows an application of this grammar in the field of automated melody harmonization.

Modern theorist Martin Rohrmeier has sought to formalize a generative grammar for all of tonal harmony. Rejecting the Markovian progression analysis of Piston and expanding upon the concept *GTTM*, "A Generative Grammar Approach to Diatonic Harmonic Structure" Rohrmeier (2007) defines a simple CFG for harmonic phrase structure. Rohrmeier's grammar can be used to create parse trees much like those found in *GTTM* (which does not present a formal grammar in and of itself); these trees can handle modulations within a close diatonic context. "Towards a Generative Syntax of Tonal Harmony" Rohrmeier (2011) expands upon his earlier grammar, indicating that rules may be added for neo-Riemannian transformations. Rohrmeier develops a syntax consisting of four levels: phrase, functional, scale degree, and surface. While Rohrmeier acknowledges that such a framework cannot sufficiently express all of the extended 19th-century chromatic idiom, he points to the possibility of using partial tree matching algorithms for such cases.

PCFG for Schubert's Phrase Structure

As a second experiment in automated phrased-based analysis, I decided to follow Rohrmeier's example in constructing a formal grammar for Schubert's harmonies. So as to provide a reliable data set, I chose to perform phrase-by-phrase Roman numeral analysis on the 24 songs from *Die Winterreise*, denoting all modulations and local tonicizations by the number of half-steps the tonic changes. (For example, a change of 2 half-steps could represent a shift from G major to A minor, and a change of 0 half-steps could represent a simple change in modality.) Using Rohrmeier's formal grammars as my inspiration, I constructed a CFG for phrase structure, adding rules until every phrase in my data set was parsable. I used an open-source Python script (currently available at https://github.com/stefanbehr/cky) to convert my grammar into Chomsky Normal Form and run the CYK parsing algorithm on the phrases (see the final section of Chapter 1 for a brief overview of these concepts). The production rules of my CFG were as follows:

1. $START \rightarrow PHRASE$
2. $PHRASE \rightarrow PHRASE\ M\ n$
3. $PHRASE \rightarrow t$
4. $PHRASE \rightarrow d\ t$
5. $t \rightarrow tp$
6. $d \rightarrow s\ d$
7. $tp \rightarrow s$
8. $t \rightarrow I$
9.-11. $tp \rightarrow \{VI,\ \flat VI,\ III\}$
12.-13. $d \rightarrow \{V,\ VII\}$
14.-18. $s \rightarrow \{IV,\ II,\ VI,\ \flat II,\ \flat VI\}$
19. $tp \rightarrow tp\ tp$
20. $tp \rightarrow t\ s\ t$
21. $tp \rightarrow t\ d\ t$
22. $t \rightarrow V/\ t\ t$
23. $t \rightarrow t\ V/\ t$
24. $tp \rightarrow V/\ tp\ tp$
25. $tp \rightarrow tp\ V/\ tp$
26. $s \rightarrow V/\ s\ s$
27. $s \rightarrow s\ V/\ s$
28. $d \rightarrow V/\ d\ d$
29. $d \rightarrow d\ V/\ d$
30.-41. $n \rightarrow \{0,\ 1,\ 2,\ \ldots,\ 11\}$

In this grammar, all Roman numerals, as well as $V/$, M, and all numbers derived from symbol n are terminals. Beginning with the unique start symbol $START$, phrases are designated as being either modulating or nonmodulating (rules 1, 2). Phrases are then classified as either tonic (t) or dominant resolution ($d\ t$) phrases. A tonic may be substituted with a tonic prolongation (tp), which can in turn be substituted with a predominant functioning chord (s). Tonic, tonic prolongation, predominant, and dominant symbols can all be ornamented by the $V/$ terminal, which indicates tonicization; rules 22-29 are included so that secondary dominants may be parsed. Tonic prolongations allow for a great deal of freedom; they may be themselves prolonged (rule 19), or turned into a local accentuation (rules 21, 22) that could otherwise be misconstrued as a complete phrase.

The terminals included in this grammar can be seen as an extension of traditional Roman numeral analysis. The t symbol can only yield I, and the d symbol can only yield V or VII. It is worth noting here that this grammar does not distinguish between major and minor modalities, including a number of options for tonic prolongation chords and relying solely on upper-case Roman numerals. The inclusion of III as a tonic prolongation can be traced to an exception Rohrmeier made in his 2007 paper; coincidentally, this exception was made with Schubert in mind. Of course, in minor keys, III is a more acceptable tonic substitute than in major keys. Denoting modulating phrases with $M\ n$ comes as a generalization of neo-Riemannian theory, foregoing specific operations on tonalities for more general shifts in tonic motion.

Having found a succinct generative grammar for Schubert's phrase structure, we may now turn our attention back to the larger question of automated, unsupervised phrase-based harmonic analysis of Schubert's songs. While I relied on performing Roman numeral analysis by hand so as to assure a trustworthy data base, Roman numeral analysis may easily be automated. Maxwell (1992) details an "expert system" for automated Roman numeral analysis that could be applied to MusicXML data. Using a series of hard-coded rules, Maxwell's algorithm views a musical score as a series of "vertical" (simultaneous) sonorities so as to solve two main problems: determining which sonorities are chords worthy of a Roman numeral, and determining the tonal region in which the chords are to be analyzed (ibid:336).

Maxwell tackles the first problem with heuristics that compare and sometimes combine adjacent events in a score. The second problem is solved by a simple parsing process that seeks out "pseudo-cadences" that can potentially identify new keys. Maxwell's algorithm can be used to convert a MusicXML score to a string of Roman numerals, with no indication of phrasing; I propose to use this method (or a related algorithm for simple Roman numeral analysis) in combination with the grammar I have created to analyze Schubert's songs on a phrase-by-phrase basis. To do so, I propose to add weights to my CFG, and to allow for parsings of entire pieces of music with the addition of one production rule:

42. *PHRASE → PHRASE PHRASE*

After re-examining my Roman numeral representations of Schubert's phrases in *Die Winterreise*, I proposed the following weights for my CFG:

RULE #	WEIGHT	RULE #	WEIGHT	RULE #	WEIGHT
1	1.000	15	0.400	29	0.450
2	0.309	16	0.100	30	0.202
3	0.700	17	0.050	31	0.023
4	0.300	18	0.050	32	0.101
5	0.700	19	0.150	33	0.132
6	0.750	20	0.300	34	0.047
7	0.450	21	0.250	35	0.140
8	1.000	22	0.250	36	0.000
9	0.900	23	0.250	37	0.101
10	0.050	24	0.350	38	0.031
11	0.050	25	0.350	39	0.093
12	0.800	26	0.350	40	0.093
13	0.200	27	0.350	41	0.038
14	0.400	28	0.450		

The weights for rules 1, 2, and 30-41 were calculated directly from the phrases in the data base. The other weights were approximated based on my own observations of Schubert's tendencies (similarly, Rohrmeier weighted his generative grammars by hand). Because the CYK parser I used relies on the grammar being in CNF, I cannot establish exact probabilities for each production rule. Regardless, this PCFG, with the addition of rule 42 (assigned a small weight so as to avoid over-segmentation of phrases), could successfully produce a phrase-by-phrase parsing of a complete song form. The weights provided may be used in the parsing process; relying on more prevalent production rules while parsing a phrase can yield well-balanced trees. The hierarchy created by such an analysis would be relatively shallow, establishing no relationship *between* phrases, but further experimentation can be done with the addition of early production rules that outline large-scale song forms. As an example, I have included a parse tree generated by this grammar for the first two phrases of "Die Krähe".
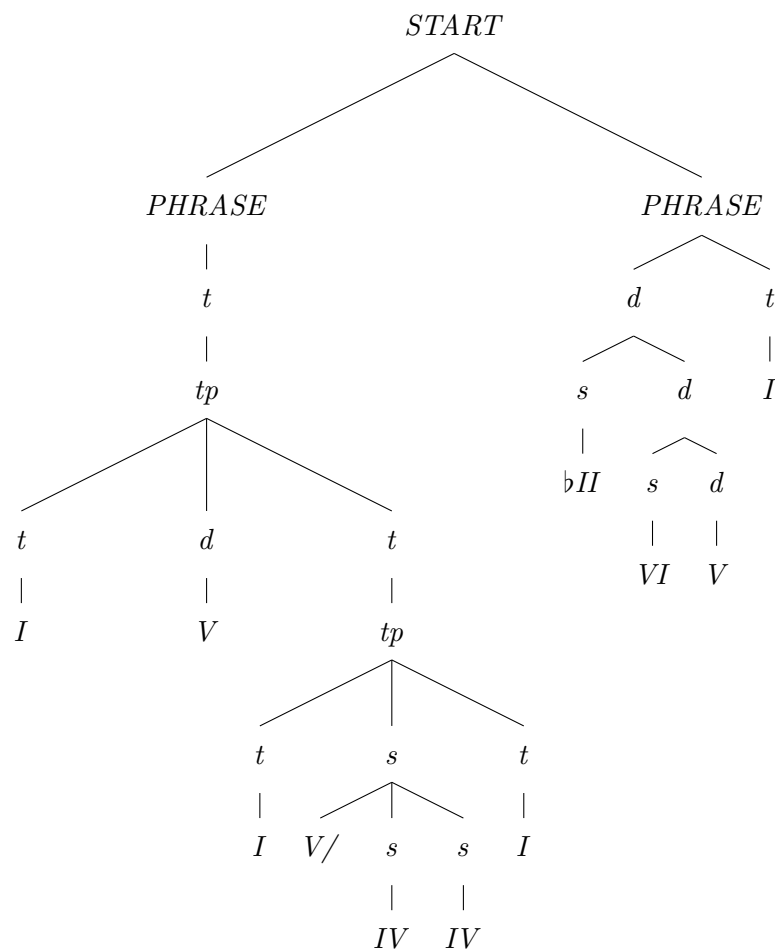
CHAPTER 3

Melodic Analysis: Self-Similarity and Motivic Development

The Issue of Similarity and Identity

This chapter is concerned with the gleaning of musical structure from a song's melody. Melodic surfaces have been analyzed in isolation to determine both local boundary points and large-scale structures via self-similarity; when considered in the context of a given harmonic framework, melodies can go far in elucidating the structure of a piece. One of my primary interests in this chapter is scripting for the discovery of methods for extracting *motives*, or salient melodic patterns that are developed/mutated throughout a piece. While the recognition of motives and their variants seems to be intuitive to the average listener (take, for example, the first four notes of Beethoven's Fifth Symphony), the automation of such recognition entails some tricky considerations. For one, there is a need to establish metrics for similarity and identity and melody: what makes two melodic fragments sound similar or distinct? While the identification of exact repeats is trivial in score parsing, amounting to mere pattern matching on prefix trees, the identification of near-identity is much more complicated.

An Overview of Unsupervised Learning in Automated Melodic Analysis

Examples of automated methods for melodic segmentation and motive analysis can be traced back to Meyer's chapter on "conformant relationships" (Meyer 1973). Meyer identifies time (temporal proximity) and pitch as the two essential elements of a motive, and argues that the key to perceiving motives lies in their individuality. To this end, he provides the following formula (ibid:49):

(strength of perceived conformance) = (regularity of pattern * individuality of profile * similarity of patterning)/(variety of intervening events * temporal distance between events)

Meyer went on to note that "on the hierarchic level where repetition is immediate, it tends to separate events. But on the next level – where similar events are grouped together as part of some larger unit – repetition tends to create coherence" (ibid:53).

The automation of Meyer's concepts of melody can be traced to Cambouropoulos (1998), a thesis presenting a new General Computational Theory of Musical Structure (*GCTMS*). The *GCTMS* is built around the concept of identity within a melody, and has the intention of supplementing theories such as the *GTTM*, which Cambouropoulos argues does not handle the motivic-thematic processes of a piece well (ibid:15). The *GCTMS* begins by defining a Local Boundary Detection Model (LBDM) for melodic segmentation, noting that segmentations need not be entirely non-overlapping (as they are in *GTTM*). The LBDM uses three simple rules to calculate changes in melodic fragments as sums (ibid:87). Given a phrase segmentation, the *GCTMS* then utilizes pattern-matching algorithms to find musical parallelism. In Cambouropoulos's phrase-based melodic analysis, each note is stored as a tuple of pitch class and metrical distance from the start of the phrase (ibid:107); this is comparable to the metrics I employ in the next section of this chapter. The final stage of the *GCTMS*'s analytical process is the application of an unsupervised UN-SCRAMBLE that iterates until it effectively categorizes (clusters) all phrases of a melody. Note that Cambouropoulos uses a rather simple metric for similarity: a category is a set of pairwise similar elements as defined by a hard threshold (ibid:120). As such, a phrase may belong to multiple categories.

A contemporary theorist, Dominik Hörnel, has researched the learning of musical structure through a combination of supervised and unsupervised algorithms. His 1996 paper "Learning Musical Structure and Style by Recognition, Prediction and Evolution" proposes MELONET II, a multi-level neural network model that incorporates the unsupervised learning of musical structure. The hierarchical clustering of motives presented in this paper served as inspiration for my work in the next section of this chapter. Hörnel and Menzel (1998) elaborate on this motivic analysis, using a simple interval representation for classifying motives as melodic (stepwise) or harmonic (chord-tone based).

Dubnov (2003) seeks to formalize Meyer's argument that expectation lies at the core of musical perception. Dubnov presents a method of parsing for motives based on pattern-matching with suffix trees and simple probability distributions for motivic occurrence (ibid:6). Pearce (2008) presents a related model for the parsing of melodic segmentations: the Information Dynamics Of Music (IDyOM) model proposes that boundaries will occur before events for which unexpectedness (information content) and uncertainty (entropy) are high (ibid:90). IDyOM relies on n-grams for its analysis, though the paper does acknowledge their limitations.

Just as formal grammars have been proposed for the parsing of tonal harmony, research has been done into the field of PCFGs for melodic motion. Bod (2001) shows that PCFGs can be learned for melodic motion in a corpus of folk songs that have been pre-processed to have segmented phrases. Bod proposes a simple Markov grammar technique that can assign a probability to any possible context-free rule, and then decompose the rule and its probability into a Markov process (ibid:5). Bod extends this model to a data-oriented parsing model that takes global considerations, such as the total number of phrases in a song, into account.

Gilbert and Conklin (2007) present a PCFG method for analyzing Bach chorale melodies (which, as noted, are pre-segmented). The rules presented in this paper are based on note-by-note melodic contour, separating "elaboration" notes from more essential notes to optimize parsing. Their method is trained with expectation-maximization, much like the Baum-Welch (forward-backward) algorithm used for HMMs.

Hierarchical Agglomerative Clustering in Motivic Analysis

As a first experiment, I implemented a simple HAC algorithm for motivic analysis. Using the songs of *Die Winterreise* as my data set, I wrote a Python script that performed the following tasks:

1. Parse through the song's melody, creating a list that logs each note encountered as a tuple of the note's pitch and its length in quarter notes.
2. Given this flattened melody, create a list of overlapping melodic n-grams, where n is a predetermined *phrase length* parameter.

3. Create and populate a *distance matrix* with the distances between all pairs of melodic n-grams. I used a simple distance metric for my first experiment: $d = \sum_{i=1}^{n} |p_{1,i} - p_{2,i}| + \sum_{i=1}^{n} |l_{1,i} - l_{2,i}|$, where p_1 is the sequence of pitches for cluster 1 and l_1 is the sequence of quarter lengths.

4. Perform HAC with a min-distance metric until the number of clusters is reduced to a pre-determined parameter k.

5. Display the elements of the largest resulting cluster.

My decision to reduce melodies to tuples of pitch and duration stems from Cope (1996:83), which asserts that these two features are fundamental to pattern-matching. My choices of n and k were somewhat arbitrary in this implementation, with n ranging from 6 to 10 and k ranging from 30 to 100 clusters. Ideally, these parameters should come as the result of a successful phrase-based harmonic analysis; once phrase boundaries are determined, the average number of notes in each phrase can easily be found, and the value of k should be a multiple of the number of phrases within the song.

The results below are promising for many of the songs, especially given the simplistic distance metric used. I notated a representative n-gram from each song's largest cluster; in many cases, motivic or otherwise memorable melodies were found. In a more integrated analysis, metrics should be devised for the melodic and rhythmic interest and variety of an n-gram, preferring (though not necessarily changing the distances between) more interesting melodies.

"Die Winterreise" Motives found by HAC

CHAPTER 4

Comparative Analysis: Hermeneutics and Feature Extraction

Feature Extraction with music21

Having discussed harmonic and melodic analysis at length, we now turn our attention to the texts Schubert set to music for the sake of comparative analysis. To supplement the data gleaned from natural language processing (discussed later in this chapter), I used music21's feature extraction module to obtain the following additional attributes of each song:

1. Amount of arpeggiation in accompaniment (continuous, $\in [0,1]$)
2. Average melodic interval (integer-valued, in half-steps)
3. Average number of independent voices (continuous)
4. Changes of meter (integer-valued)
5. Direction of melodic motion (continous, $\in [0,1]$, 1 for entirely upward motion)
6. Range (integer-valued, in half-steps)

Cuthbert (2011) shows how these and related features can be used to distinguish songs from diverse sources; they can yield particularly effective results when fed to a binary classifier. de Haas (2009) points to a direction for future research: by using pattern-matching on parse trees produced by Rohrmeier's generative grammar, a similarity matrix may be constructed to compare songs on the basis of their chord progressions. The results of my data analysis using the features I extracted can be found later in this chapter.

LDA for Sentiment Analysis

It has been argued that Schubert's musical style in his songs is largely a function of his choices of text; Clark (2011:55) states that Schubert's song structures and affectations are largely dictated by the text. To this end, I decided to experiment with Latent Dirichlet Allocation (explained in Chapter 1) to gain insight into Schu-

bert's textual "topics" as they relate to his musical compositions. After collecting the texts for the 107 songs in my corpus, I scraped them of all punctuation, viewing each as a "bag of words" with associated frequencies. I created a vocabulary of all words appearing at least 3 times and at most 100 times in the corpus; this removed extraneous words from consideration while simultaneously preventing the formation of topics based on words that are common for grammatical reasons (articles, pronouns, etc.). I ran Blei's C implementation of LDA (currently available at http://www.cs.princeton.edu/~blei/lda-c/), trying several parameters and settling on running the algorithm with 6 latent topics. By analyzing the songs most closely associated with each topic in conjunction with the top words "generated" by each topic, some clear groupings arise.

The first topic is associated with the words "Seelen" (souls), "Frieden" (peace), and "ruhen" (rest), and with the songs "An die Freude" (Ode to Joy), "Am Tage aller Seelen" (On All Souls' Day), "Lob der Tränen" (In Praise of Tears), "Der Einsame" (The Lonely), and "Trockne Blumen" (Dry Flowers). These songs are alike in their deeply peaceful, sweet, and reflective sentiments (Reed 1997:96,323,324).

The second topic is associated with the words "muß" (must), "Nacht" (night), "Liebe" (love), and "wieder" (back/again); it is associated with the songs "Alinde", "Gute Nacht" (Good Night), "Der Hirt auf dem Felsen" (the Shepherd on the Rock), and "Des Baches Wiegenlied" (The Creek's Lullaby). These songs are alike in their dark, stoic, and dramatic natures (Reed 1997:13,107,192,444).

The third topic is associated with the words "wohl" (well), "Herz" (heart), and "Freund" (friend), and with the songs "Ave Maria", "Im Frühling" (In Springtime), "Frühlingstraum" (Dream of Spring), and "Abschied" (Farewell). These songs are linked by pleasant aestheticism that is not without a disillusioned undertone (Reed 1997:217,279,365,450).

The fourth topic is associated with the words "Liebe", "meine" (mine), "Vater" (father), and "Röslein" (little rose); it is linked with the songs "Heidenröslein" (Little Rose on the Heath), "Erlkönig" (The Erlking), "Der Zwerg" (The Dwarf). "Heidenröslein" and "Erlkönig" are alike in that they are both texts by Goethe about the destruction of innocence (Byrne 2003:209,222); "Der Zwerg" has been compared with "Erlkönig" due its similar nature and through-composition (Reed

1997:143).

The fifth topic generates the words "Welt" (world), "Land" (country), "Herz", and "mein"; it is associated with the songs "Sei mir Gegrüßt", (Greet Me) "Die böse Farbe" (The Evil Color), "Der Wanderer" (The Wanderer), and "Der Rattenfänger" (The Pied Piper). This topic shows the least thematic consistency among the 6; the words generated suggest patriotism, but the songs associated with it are largely disparate.

The sixth and final topic generates the words "Wandern" (wander), "Tränen" (weep), "Herz", and "Bächlein" (little brook). It is linked with the songs "Ungeduld" (Impatience), "Gretchen am Spinnrade" (Gretchen at the Spinning Wheel), "Frühlings-sehnsucht" (Longing in Spring), and "Die Taubenpost" (The Pigeon Post). This topic is among the most consistent in its sentiment; the songs are all linked by a feeling of *Sehnsucht*, or Romantic longing. Sehnsucht is expressed in these songs through rhythmic irregularities (Malin 2006) (Reed 1997:185) (Byrne 2003:110); the songs' related musical affectations can be traced back to their accompanying texts.

For an unsupervised, generative algorithm, LDA performed surprisingly well given the relatively small size of the data set. By searching for topics in Schubert's vocabulary, a good deal can be learned about the underlying sentiments of his songs and their texts.

Schubert's Poets, Tonalities, and Chronology

With a database of song features and LDA topic associations, I logged each song's poet, (starting) key, and year of composition in an attempt to find trends over Schubert's career. The data yielded the following insights:

BY POET:

- Goethe's poems, on average, were most closely associated with the fourth LDA topic ($\overline{\gamma_4'} = 27.17$ for Goethe's songs, with $\overline{\gamma_4} = 8.37$ for the entire data set), with the sixth topic being the next closely related ($\overline{\gamma_6'} = 19.22$; $\overline{\gamma_6} = 10.72$). Note that the variable γ represents the weight assigned to a given topic for a text. These results are in line with Goethe's generally dark and contemplative

texts. On a related note, Schubert's tendency to through-compose Goethe's texts yielded high average numbers of meter changes ($\overline{n'} = 0.60$ for Goethe; $\overline{n} = 0.32$ for all songs) and independent voices ($\overline{m'} = 2.15$; $\overline{m} = 1.65$).

- The poems of Müller, whose texts make up the song cycles *Die schöne Müllerin* and *Die Winterreise*, have a fairly uniform distribution across topics: ($\overline{\gamma_1'} = 2.47$, $\overline{\gamma_2'} = 11.77$, $\overline{\gamma_3'} = 10.37$, $\overline{\gamma_4'} = 7.70$, $\overline{\gamma_5'} = 10.43$, $\overline{\gamma_6'} = 10.03$), with ($\overline{\gamma_1} = 9.31$, $\overline{\gamma_2} = 10.97$, $\overline{\gamma_3} = 8.56$, $\overline{\gamma_4} = 8.37$, $\overline{\gamma_5} = 7.49$, $\overline{\gamma_6} = 10.72$). This shows the complex range of emotions displayed in the poet's work.

BY YEAR:

- 1815, a very prolific year for Schubert marked by his first widespread acclaim (Byrne 2003:28), yielded songs closely associated with the first ($\overline{\gamma_1'} = 30.89$; $\overline{\gamma_1} = 9.31$) and fourth ($\overline{\gamma_4'} = 33.20$; $\overline{\gamma_4} = 8.37$) topics, showing Schubert's more happy sentiments in his earlier compositions and his increasing interest in storytelling through song, respectively. The songs of 1815 also had, on average, small ranges in their instrumentation ($\overline{r'} = 47.43$; $\overline{r} = 52.43$), a feature that generally increased over the course of Schubert's career.

- 1827, the year of *Die Winterreise*, has a fairly uniform distribution of topics, but is most closely related to the second (dark, stoic) topic ($\overline{\gamma_2'} = 12.46$; $\overline{\gamma_2} = 10.97$). 1828, the year of Schubert's death and the publication of his posthumous *Schwannengesang*, was most closely associated with the second ($\overline{\gamma_2'} = 11.46$; $\overline{\gamma_2} = 10.97$) and sixth ($\overline{\gamma_6'} = 16.86$; $\overline{\gamma_6} = 10.72$) topics, a sign of Schubert's gloomy resolve as he fought his final illness. This year had the greatest average instrumental range ($\overline{r'} = 63.2$; $\overline{r} = 52.43$) and large vocal intervals ($\overline{v'} = 4.53$; $\overline{v} = 4.07$).

BY KEY:

- D minor, like the year 1828, is strongly associated with the second ($\overline{\gamma_2'} = 26.02$; $\overline{\gamma_2} = 10.97$) and sixth ($\overline{\gamma_6'} = 17.62$; $\overline{\gamma_6} = 10.72$) topics. Schubert's D minor songs were often "powerful dramatic masterpieces" about "man's courage and resolution in a struggle against fate" (Reed 1997:485-486).

- A minor is very closely associated with the fourth topic ($\overline{\gamma_4'} = 29.27$; $\overline{\gamma_4} = 8.37$), with predominantly downward ($\overline{m'} = 0.46$) melodic motion. Schubert's use of A minor "is firmly associated with disenchantment, alienation, and derangement" (ibid:489).

- E major is very strongly associated with the first topic ($\overline{\gamma_1'} = 86.62$; $\overline{\gamma_1} = 9.31$), representing songs filled with "innocence and joy" (ibid:486).

The distributions of LDA topics across keys, poets, and years point to a composer who chose his texts carefully and deliberately, writing songs that heightened these texts through musical affect. LDA performed very well in grouping Schubert's songs by the sentiments of their texts; while it is difficult to directly measure the affect of a song as a musical composition, examining its various features in combination can yield useful new insights.

Conclusions

The methods presented in this paper are intended to be used not in lieu of traditional musical analysis, but as a supplemental toolkit for musicologists. By viewing the melody, harmony, and text of a song as three orthogonal components, we are allowed to focus on formal structure, but this is no substitute for the holistic examination of the piece's aesthetic beauty and affect. Given a score and its accompanying text, the methods presented here provide a snapshot of Schubert's compositional process, illuminating phrase structure, melodic development, and the sentiment of the chosen text.

I chose Franz Schubert as my subject because critics to this day view his songs as defying analysis (Reed 1997:182), composed by a somnambulistic recluse who chose his texts indiscriminantly (Byrne 2003:30). Believing that Schubert's compositions had to fit into some kind of consistent extended tonal framework, I sought out generative rules for Schubert's harmonies and choices of text. The results pointed to several prevalent trends in both Schubert's phrase structures and his song-to-song compositional choices.

ACKNOWLEDGEMENTS

Bibliography

Adán, Víctor Gabriel. "Hierarchical Music Structure Analysis, Modeling and Resynthesis: A Dynamical Systems and Signal Processing Approach." MIT, 2005.

Blei, David M. et al. "Latent Dirichlet Allocation." *Journal of Machine Learning Research 3*, 2003.

Bod, Rens. "Probabilistic Grammars for Music." University of Amsterdam, 2001.

Burkholder, J. Peter et al. "A History of Western Music (Eighth Edition)." Norton: New York, 2010.

Byrne, Lorraine. "Schubert's Goethe Settings." MPG Books: Cornwall, 2003.

Cambouropoulos, Emilios. "Towards a General Computational Theory of Musical Structure." University of Edinburgh, 1998.

Chai, Wei. "Automated Analysis of Musical Structure." MIT, 2005.

Clark, Suzannah. "Analyzing Schubert." Cambridge UP: New York, 2011.

Cohn, Richard. "As Wonderful as Star Clusters: Instruments for Gazing at Tonality in Schubert." *19th-Century Music*, Vol. 22, No. 3, 1999.

Cohn, Richard. "The Autonomy of Motives in Schenkerian Accounts of Tonal Music." *Music Theory Spectrum*, Vol. 14, No. 2, 1992.

Cope, David. "Experiments in Musical Intelligence." A-R Editions: Madison, 1996.

Cuthbert, Michael, et al. "Feature Extraction and Machine Learning on Symbolic Music Using the music21 Toolkit." *The International Society for Music Information Retrieval (ISMIR)*, 2011.

Dubnov, Shlomo et al. "Using Machine-Learning Methods for Musical Style Modeling." *IEEE Computer Society*, 2003.

Everett, Walter. "Grief in 'Winterreise': A Schenkerian Perspective." *Music Analysis*, Vol. 9, No. 2, 1990.

Gilbert, Edouard and Darrell Conklin. "A Probabilistic Context-Free Grammar for Melodic Reduction." Ecole Normale Superieure de Cachan, 2007.

de Haas, W. Bas et al. "Modeling Harmonic Similarity Using a Generative Grammar of Tonal Harmony." *ISMIR*, 2009.

Hofstadter, Douglas R. "Godel, Escher, Bach: An Eternal Golden Braid."
    Vintage: New York, 1979.

Hörnel, Dominik and Wolfram Menzel. "Learning Musical Structure and Style with
    Neural Networks." *Computer Music Journal*, Vol. 22, No. 4, 1998.

Hörnel, Dominik and Thomas Ragg. "Learning Musical Structure and Style by
    Recognition, Prediction, and Evolution." *ICMC Proceedings*, 2006.

Jackendoff, Ray. "Musical Parsing and Musical Affect." *Music Perception:
    An Interdisciplinary Journal*, Vol. 5, No. 3, 1988.

Kirlin, Phillip B. and David D. Jensen. "Probabilistic Modeling of Hierarchical
    Music Analysis." *ISMIR*, 2011.

Koops, H. V. "A Model Based Approach to Automatic Harmonization of a Melody."
    Utrecht University, 2012.

Lerdahl, Fred and Ray Jackendoff. "A Generative Theory of Tonal Music."
    MIT Press: Cambridge, 1983.

Lerdahl, Fred. "Tonal Pitch Space." *Music Perception: An Interdisciplinary
    Journal*, Vol. 9, No. 2, 1991.

Magalhaes, Jose Pedro and W. Bas de Haas. "Functional Modelling of Musical
    Harmony." Utrecht University, 2011.

Malin, Yonatan. "Metric Displacement Dissonance and Romantic Longing in the
    German Lied." Wesleyan University, 2006.

Maxwell, H. John. "An Expert System for Harmonizing Analysis of Tonal Music."
    Understanding Music with AI: Perspectives on Music Cognition.
    MIT: Cambridge, 1992.

Meyer, Leonard B. "Explaining Music." University of California: Berkeley, 1973.

Palmer, Caroline. "Music Performance." *Annual Review of Psychology*,
    Vol. 48, 1997.

Pearce, M.T. et al. "A Comparison of Statistical and Rule-Based Models of Melodic
    Segmentation." *ISMIR*, 2008.

Piston, Walter. "Harmony." Norton: New York, 1941.

Ponsford, Dan et al. "Statistical Learning of Harmonic Movement." University
    of Edinburgh, 1999.

Raphael, Christopher and Joshua Stoddard. "Functional Harmonic Analysis
Using Probabilistic Models." *Computer Music Journal*, Vol. 28, No. 3, 2004.

Reed, John. "The Schubert Song Companion." Mandolin: Manchester, 1997.

Rohrmeier, Martin. "A Generative Grammar Approach to Harmonic Structure."
SMC Proceedings, 2007.

Rohrmeier, Martin. "Towards a Generative Syntax of Tonal Harmony." *Journal of
Mathematics and Music*, Vol. 5, No. 1, 2011.

Schachter, Carl. "Unfoldings: Essays in Schenkerian Theory and Analysis."
Oxford UP: New York, 1999.

Taube, Heinrich. "Automatic Tonal Analysis: Toward the Implementation of a
Music Theory Workbench." *Computer Music Journal*, Vol. 23, No. 4, 1999.

Tymoczko, Dmitri. "A Geometry of Music: Harmony and Counterpoint in the
Extended Common Pracitce." Oxford UP: New York, 2011.

Winograd, Terry. "Linguistics and the Computer Analysis of Tonal Harmony."
Journal of Music Theory, Vol. 12, No. 1, 1968.