

A NEW DATASET, NOTATION SOFTWARE, AND REPRESENTATION FOR COMPUTATIONAL SCHENKERIAN ANALYSIS

Stephen Ni-Hahn¹ Weihan Xu¹ Jerry Yin¹ Rico Zhu¹
Simon Mak¹ Yue Jiang¹ Cynthia Rudin¹

¹Duke University, USA

stephen.hahn@duke.edu

ABSTRACT

Schenkerian Analysis (SchA) is a uniquely expressive method of music analysis, combining elements of melody, harmony, counterpoint, and form to describe the hierarchical structure supporting a work of music. However, despite its powerful analytical utility and potential to improve music understanding and generation, SchA has rarely been utilized by the computer music community. This is in large part due to the paucity of available high-quality data in a computer-readable format. With a larger corpus of Schenkerian data, it may be possible to infuse machine learning models with a deeper understanding of musical structure, thus leading to more “human” results. To encourage further research in Schenkerian analysis and its potential benefits for music informatics and generation, this paper presents three main contributions: 1) a new and growing dataset of SchAs, the largest in human- and computer-readable formats to date (>140 excerpts), 2) a novel software for visualization and collection of SchA data, and 3) a novel, flexible representation of SchA as a heterogeneous-edge graph data structure.

1. INTRODUCTION

With the continuously growing availability of “big data,” machine learning models and algorithms have made enormous strides in many fields, such as computer vision and language modeling. Recent approaches to music information retrieval (MIR) and music generation tasks are increasingly fueled by massive datasets as well, particularly when working with raw audio. For instance, for generation tasks, Meta’s *MusicGen* is trained on approximately 20,000 hours of licensed music [1], OpenAI’s *Jukebox* on 1.2 million songs [2], and Google’s *Noise2Music* on 340,000 hours of music [3]. Castellon et al. show how these large generation models produce useful representations for downstream MIR tasks [4]. Won et al. perform

multimodal metric learning for tag-based music retrieval using approximately 158,000 tracks [5].

Despite this promising body of work, many areas of music research do not have access to such data and are therefore under-researched and underappreciated, particularly in the realm of symbolic music or Schenkerian Analysis (SchA). By infusing an understanding of Schenkerian musical structure, generative machine learning models may be able to learn more artistic, theoretically-informed structural features beyond simple form and metric features when making inference. Unfortunately, there is currently only one sizeable publicly available dataset for SchA in computer-readable format, and it is relatively small with 41 excerpts [6].

Schenkerian analysis provides a powerful, flexible, and broadly-used analytical framework for understanding musical melodic-harmonic structure in a sensitive, “human” way. Rather than viewing a piece of music as a series of vertical chunks or horizontal melodies, SchA instead analyzes music as an artistic “unfolding” of harmony through time, taking into account elements of melody, harmony, form, and counterpoint. Schenker’s theories have inspired numerous performers and composers [7–9], helping them to understand their own interpretations of musical structure, which in turn may inform their own performance and/or composition. An understanding of Schenkerian structure helps performers determine what notes deserve emphasis and which may be more transient. A composer can learn to imitate and develop structures they have seen in other pieces of music they admire.

Because Schenkerian theory requires a significant amount of background knowledge in music theory and practice and has a difficult learning curve, it is often overlooked or misunderstood. For instance, SchA is often deemed too narrow due to its origins in repertoire of Western common practice tonal music. However, aspects of Schenkerian theory have shown strong analytical power in works of popular, rock, jazz, and even African folk music, Chinese opera, and 20th century western atonal music [10–13]. To be clear, we see SchA as a broad and evolving field with various analytical tools that can be applied to a wide array of musical genres, not as a static theory solely designed for common practice tonality.

It is our belief that research in computational SchA can enable performers and composers to more easily analyze music and guide the process of understanding mu-



© S. Ni-Hahn, W. Xu, J. Yin, R. Zhu, S. Mak, Y. Jiang, and C. Rudin. Licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0). **Attribution:** S. Ni-Hahn, W. Xu, J. Yin, R. Zhu, S. Mak, Y. Jiang, and C. Rudin, “A New Dataset, Notation Software, and Representation for Computational Schenkerian Analysis”, in *Proc. of the 25th Int. Society for Music Information Retrieval Conf.*, San Francisco, United States, 2024.

sical structure. Computational SchA can also aid the expert human analyst by offering several potential solutions, speeding up their ability to parse through a piece of music or potentially unveiling unusual and interesting analytical insights. The computer would not replace the human expert; rather, it would help the analyst find reasonable solutions more quickly, which would be immensely helpful when conducting large-scale corpus studies. Furthermore, inclusion of SchA in MIR and generation tasks may greatly improve results. This injection of computational models with musical theory and/or structure has shown benefits in numerous MIR and generation tasks [14–17].

This paper introduces three main contributions. The first is a new and growing **SchA dataset**, the largest in human- and computer-readable format to date (>140 excerpts). Second, we present a novel **notation software for SchA** in an effort to ease data collection and visualization. Lastly, we describe a **representation of SchA** as a graphical data structure and graph pooling problem.

The following subsections describe SchA in more detail, as well as the relevant history of computational SchA. Section 2 describes our novel dataset and data collection tool. Finally, Section 3 describes how SchA may be represented as a graph data structure.

1.1 Hierarchical Music Analysis

Music is often composed and understood in terms of hierarchical structures such as phrase and rhythmic structure [18, 19], form structure [20, 21], and linear/harmonic structure [22, 23]. In this paper, we focus on the Schenkerian model of harmonic-melodic structure. As mentioned earlier, SchA aims to reveal how harmonies are “unfolded” through linear motion on various levels of structure. Figure 1 shows the relationship between a fugue’s subject melody and its underlying harmony, as well as the hierarchy of melodic tones.

Figure 1: The primary author’s analysis of J.S. Bach’s F major fugue subject from *Das Wohltemperierte Klavier I*.

The annotated score on the upper line shows how notes relate on various levels of structure, forming two theoretical outer voices. Longer stems indicate deeper levels of structure. The reduction on the bottom line exemplifies the underlying harmony that is unfolded by the subject melody. Green-stemmed notes correspond with the deep outer voices of the reduction.

SchA has shown that similar harmonic and motivic features often exist on multiple levels of hierarchy, revealing music’s “fractal” nature [24]. For instance, in Figure 1, the foreground melody within the first full measure (D4-C4-Bb3) can be seen as a *parallelism* of the deep level melody spanning the entire excerpt (C4-Bb3-A3); the two melodies have a similar motivic descending third in step-wise motion. One can also see the first full measure leading into the second measure as a $V^b - I$ motion in the key of V, paralleling the deep level $V^b - I$ shown in the reduction. While these examples are on a very small scale, one can see more complex harmonic and motivic structures unfolded through entire pieces. For instance, see Example 12 in [24] describing Schubert’s *Erlkönig* or Example 2 in [25] describing The Beatles’ *Something*.

Because these same music-theoretical ideas and motifs permeate multiple levels of structure, the use of a carefully-designed machine learning model may reveal such structure in a layered approach. With the rise of machine learning in data science, this calls into importance the need for computer-readable SchA datasets for model training.

1.2 Previous Work and Data for Computational Schenkerian Analysis

The majority of past attempts at computational SchA [26–30] were based on heuristics and rule-based algorithms, and therefore did not require a true computer-readable dataset for SchA. To our knowledge, Marsden [31] was the first to venture towards a machine learning approach, using a humble corpus of six Mozart analyses. He developed a “goodness metric” based on these six analyses to find the best candidate analyses within a massive search space.

More recently, Kirilin designed a probabilistic model for SchA that understands SchAs as maximal outerplanar graphs (MOPs) and learns how likely certain notes *prolong* others using random forests [32, 33]. He defines prolongation as “a situation where an analyst determines that a group of notes is elaborating a group of more structurally fundamental notes.” For instance, the syntax follows the pattern $X(Y)Z$, where the note(s) of Y prolong the motion from note X to note Z .

One potential drawback of Kirilin’s model is that it always considers one musical voice as one theoretical voice. Looking back at Figure 1, for example, we see there is clearly a deep level bass motion from E3 to F3, supporting the upper voice, which follows the motion C4-Bb3-A3. The sixteenth notes of m. 2 act to fill the gap between the lower and upper theoretical voices. An MOP reduction of the melody would force all notes to be understood as a single theoretical voice, thus obscuring the underlying counterpoint of the passage. For this reason, we represent SchA as a more general graph data structure, described further in Section 3.

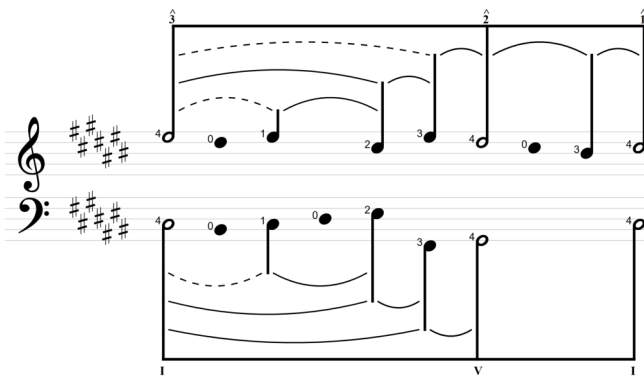
For his model, Kirilin released the first large-scale computer-readable dataset of symbolic music with corresponding expert Schenkerian analyses, *Schenker41* [6]. This collection consists of 41 excerpts from the common practice period of European art music, with analyses from

```

trebleNotes: {
  pitchNames: [n.E5, n.D5, n.E5, n._, n.C5, n.E5, n.D5, n.C5, n.B4, n.C5],
  depths: [4, 0, 1, 0, 2, 3, 4, 0, 3, 4],
  selected: [],
  ursatz: [0, 6, 9],
  scaleDegree: [3, 0, 0, 0, 0, 0, 2, 0, 0, 1],
  flagged: [],
  sharps: [],
  flats: [],
  naturals: [],
  parenthetical: []
},
bassNotes: {
  pitchNames: [n.C3, n.B2, n.C3, n.D3, n.E3, n.F2, n.G2, n._, n._, n.C3],
  depths: [4, 0, 1, 0, 2, 3, 4, 0, 0, 4],
  selected: [],
  ursatz: [0, 6, 9],
  scaleDegree: [1, 0, 0, 0, 0, 0, 5, 0, 0, 1],
  flagged: [],
  sharps: [],
  flats: [],
  naturals: [],
  parenthetical: []
},

```

(a) JSON representation.



(b) Graphical representation.

Figure 2: Screenshots of a toy Schenkerian analysis in JSON and graphical form as generated by our notation software.

three textbooks [23, 34–36] and an independent, anonymous expert in Schenkerian analysis. Kirilin also created the first computer-readable format for Schenkerian analysis, which describes all prolongations present in an analysis. The text-based format can also encode linear progressions, omitted repetitions, and harmonic context.

The Schenker41 dataset is an important first step towards broader musical-hierarchical research in the MIR community; however, there are some limitations. First of all, the quality of the excerpts chosen are questionable. Kirilin and Jensen recruited three expert Schenkerian analysts to grade textbook analyses as well as their machine learning model’s analyses in their 2015 paper (see Figure 8 in [37]). One would expect the textbook analyses to receive grades of “A-” or greater, allowing wiggle room for differences of opinion. However, many excerpts score lower marks; some were even given failing grades. Given the high proportion of dubious quality “ground truth” data, it is necessary to produce a greater quantity of quality data before successful, generalizable models can be trained.

There are also several Schenkerian symbols and concepts that are not currently represented in the text-based notation. For instance, unfoldings, voice exchanges, and other hierarchical harmonic function information are ignored. Concerning larger pieces, it is vital to understand the harmonic structure in several layers; an F major triad may stand as a local tonic “I” harmony in the foreground that serves to expand a deeper subdominant “IV” of the background, global key of C. Furthermore, more abstract concepts, such as motivic parallelisms, implied tones, and written commentary are eschewed for the sake of simplicity.

2. DATASET AND NOTATION SOFTWARE FOR SCHENKERIAN ANALYSIS

We thus introduce a new large-scale dataset of Schenkerian analyses in human- and computer-readable formats. As of the writing of this paper, the dataset contains 145 analyses from four analysts for a broad range of composers including J.S. Bach, Mendelssohn, Brahms, Bartók, Shostakovich, Gentle Giant, and more. The majority of

analyses were created by the first author (Stephen Ni-Hahn) with consultation from one of the other analysts, who wishes to remain anonymous at this time. Ni-Hahn currently has nearly a decade of experience with SchA including a graduate degree in music theory. The other three analysts are veteran Schenkerian scholars with several decades of experience in the field. The dataset is not static and aims to grow over time. Please contact Stephen Ni-Hahn (stephen.hahn@duke.edu) for questions regarding, and access to the dataset and notation software described in this paper.

Currently, the vast majority of analyses in the dataset describe the hierarchical relationships within fugue subjects by Bach and Pachelbel. Fugue subjects are ideal for preliminary trials with machine learning models since subjects are generally brief, consist of a single instrumental line (which may consist of multiple theoretical voices), generally have clear functional relationships, and each have a definite sense of closure by their end.

Rather than writing out each prolongation explicitly, we produce prolongations as a by-product when assigning a hierarchical *depth* to each note. For example, Figure 2 shows a toy example of an analysis in which the numbers to the left of the note heads indicate depth. Higher depth indicates deeper structure. To retrieve the prolongations, we simply traverse the graph at each depth level (greater than 0), connecting consecutive notes that are at the same level or higher. Custom prolongations that do not occur within this system may be added in a similar fashion to Kirilin’s text format by describing the voice and index of the start, middle, and end notes.

Figures 3 and 4 present simple statistics about our dataset. Figure 3a shows the distribution of excerpt lengths in terms of *verticalities*. A verticality is defined as a point in time where one or both of a treble and bass note exist. Note that this does not measure length of time or number of measures; rather, the number of verticalities describes the number of potentially unique depths in an excerpt. Figure 4 shows the distribution of intervals between consecutive notes in the treble and bass voices at various depths. We see that as depth increases, the distribution of treble in-

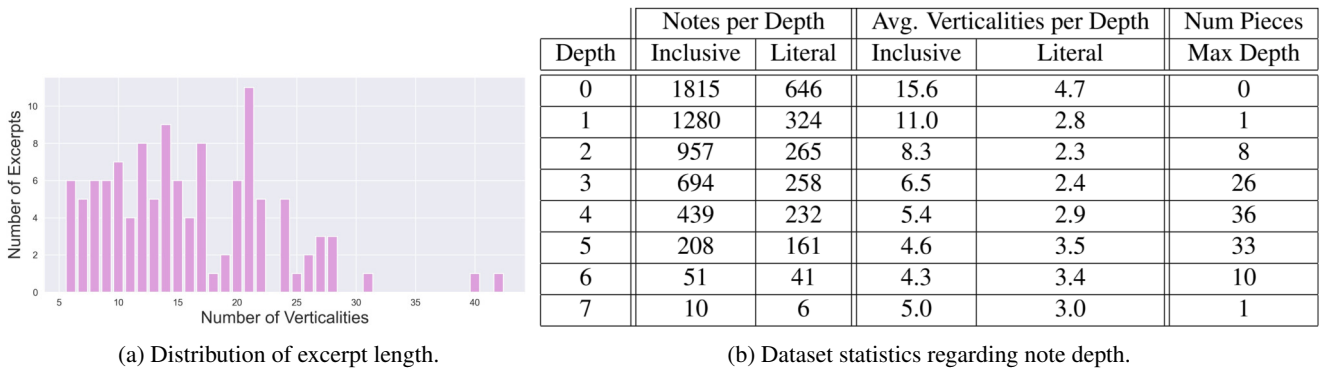


Figure 3: Dataset statistics. Verticality is defined as a point in time where one or both of a treble and bass note exist. “Inclusive” includes notes of higher depth when counting notes of lower depths. “Literal” counts the note depths as they are defined. The final column describes the distribution of max depths over all excerpts. See Section 2 for more details.

Intervals moves from smaller to larger intervals, while bass intervals increasingly concentrate around 0 and 5. These statistics suggest that surface level treble motions in our dataset are mostly stepwise and span larger intervals at deeper levels of structure. Furthermore, deep bass structures tend to hold steady and support the upper voice or move along the circle of fifths by jumping 5 or 7 half steps. Table 3b describes various statistics regarding the notes and depths of our dataset. Columns labeled “inclusive” mean that notes of higher depth are included when counting notes of lower depths. For instance, a depth 4 note is counted in the number of depth 0 notes, while the depth 4 note would not count towards the number of depth 5 notes. The “literal” label counts the note depths as they are defined. The final column describes the distribution of max depths over all excerpts.

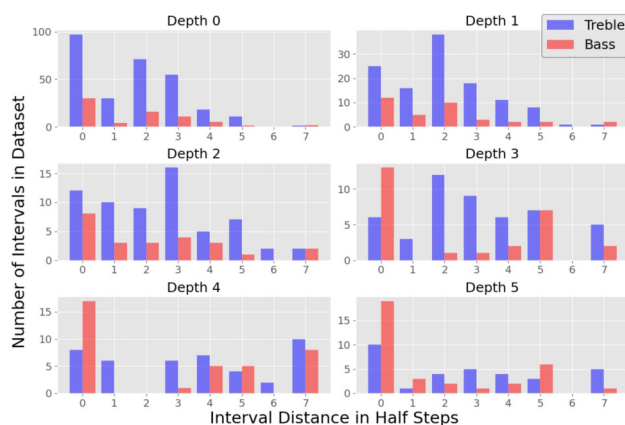


Figure 4: Distribution of intervals between consecutive notes at each depth.

2.1 Data Collection Tool

To facilitate easy collection and visualization of Schenkerian data, we introduce a new computer notation system for Schenkerian analyses (see Figure 2 for a screenshot).

As of the writing of this paper, our software is capable of notating up to four voice structures of any length. Simple commands allow the user to adjust the pitches, note depths, harmonic/scale-degree label, notes considered part

of the *Ursatz*, etc. Slurs and beams of the outer theoretical voices are automatically generated based on the depths of the notes. We are currently working on ways to render custom markings, such as voice exchanges, unfoldings, and linear progression beams.

Behind the scenes, the Schenkerian analysis is a simple standardized object in JavaScript Object Notation (JSON), which is *highly generalizable, lightweight, and simple to parse*, and is capable of describing any obscurities within a particular analysis. Our JSON object contains metadata about the analysis, key information, and information on each of four theoretical voices. Metadata describes the analyst, composer, title, subtitle, and any associated written description of the analysis. Furthermore, each theoretical voice is encoded as a list of pitch names, depths, *Ursatz* indices, scale degree/Roman numerals, flagged note indices, sharp/flat/natural indices, and parenthetical indices. Additionally, the JSON object stores “cross voice” symbols such as voice exchange lines and lines indicating related tones across larger spans of time.

Note that it is straightforward to translate between Kirlin’s OPC text notation and our JSON notation. To translate from text to JSON, the notes can be parsed from the musicxml and placed in their appropriate voice. Then note depths may be determined by the location and relative length of their prolongation. Translating from JSON to text is simpler, as one can traverse each depth and retrieve the prolongations.

The software is constructed using languages Javascript/Typescript and the Vue web framework. It is packaged using Electron Forge. Software access can be requested by emailing the first author.

3. SCHENKERIAN ANALYSIS AS A HETEROGENEOUS GRAPH DATA STRUCTURE

As mentioned in Section 1.2, Kirlin’s model simplifies the difficult problem of performing SchA, using a limited version of Yust’s MOP representation for SchA. With a greater amount of data, less compromising representations may be used for modeling. The following section describes how a musical score may be represented as a

algo2e 1 JSON to Clusters

Definitions

$parts \leftarrow \{sop, alto, ten, bass\}$
 $n_v \leftarrow$ the number of verticalities v (indexed by i) in an analysis
 $p_i \leftarrow$ note of part $p \in parts$ within v_i
 $d_i^{(p)} \leftarrow$ depth of note p_i
 $\forall p \in parts, \text{len}(p) = \text{len}(d^{(p)}) = n_v.$

Procedure CLUSTER(p, i)

```

if  $\exists j < i$  s.t.  $d_j^p > 0$  then
     $j \leftarrow \underset{j}{\text{argmin}} |i - j|$  s.t.  $j < i$  and  $d_j^{(p)} > 0$ 
    return  $\{(p, j)\}$  // Note in the same voice to the left
else if  $\exists j > i$  s.t.  $d_j^p > 0$  then
     $j \leftarrow \underset{j}{\text{argmin}} |i - j|$  s.t.  $j > i$  and  $d_j^{(p)} > 0$ 
    return  $\{(p, j)\}$  // Note in the same voice to the right
else
     $j_1, j_2 \leftarrow \underset{j_1, j_2}{\text{argmin}} \min(|i - j_1|, |i - j_2|)$  s.t.  $(i - j_1) \cdot (i - j_2) \leq 0$  and  $d_{j_1}^{(sop)} > 0$  and  $d_{j_2}^{(bass)} > 0$ 
    return  $\{(sop, j_1), (bass, j_2)\}$  // Closest two notes in outer voices in opposite directions to the inner voice note
end if
    
```

heterogeneous-edge directed graph data structure and how SchA may be conceptualized as a graph clustering problem.

3.1 Graph Music Representation

In what follows, we represent music as a heterogeneous directed graph G , where each node describes a note, and various types of edges describe the relationships between notes. Concretely, G is represented as (\mathbb{A}, X) , where $\mathbb{A} \in \{0, 1\}^{h \times n \times n}$ describes the set of h adjacency matrices (one for each edge type) over n nodes, and $X \in \mathbb{R}^{n \times d}$ is the node feature matrix with d as the number of features. These d features may be learned by a neural network, for instance, to correspond with categorical and numerical musical features.

We adapt the encoding scheme proposed by Jeong et al. [38] for the purpose of Schenkerian analysis. Nodes may be encoded with any musical feature present in the score, such as pitch class, octave, absolute duration, position (absolute or relative), metric strength, etc. We suggest the use of five main edge types: (i) forward edges connect two consecutive notes within a voice, (ii) onset edges connect notes that begin at the same time, (iii) sustain edges connect notes that are played while the source note is held, (iv) rest edges are like forward edges, but imply a rest occurs between the two related notes, and (v) linear edges connect each note with the next notes that occur at specific intervals from the source.

3.2 Schenkerian Analysis as Hierarchical Clustering

With this graphical representation of music, the process of Schenkerian analysis may then be posed as a hierarchical graph clustering problem. Figure 5 presents a toy example of how Schenkerian analysis may be represented as a series

of hierarchical clusters. The clustering between two subsequent levels of Schenkerian analysis is expressed through a *clustering matrix*, $S^{(l)} \in \mathbb{R}^{n_l \times n_{l+1}}$, where n_l is the number of nodes in *clustering layer* l and $n_{l+1} < n_l$ is the number of nodes after one iteration of clustering. We define n_0 to be the total number of notes in the music.

Note that we can understand a clustering between *any* two layers as a single matrix, denoted as $S^{(l_i) \rightarrow (l_j)} \in \mathbb{R}^{n_{l_i} \times n_{l_j}}$; $i < j$, where i and j are the index of the source and destination layers respectively. This single matrix is obtained by simply multiplying all sequential clustering matrices. For example, in Figure 5, to retrieve the matrix describing how all five nodes of the original score are clustered into the two nodes of the final middleground layer, we can multiply each clustering matrix together:

$$S^{(0) \rightarrow (2)} = S^{(0)} \cdot S^{(1)} \cdot S^{(2)} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}^\top.$$

3.3 Converting Schenkerian Analyses from JSON to Matrix Notation

Schenkerian analysis JSON data (collected using our tool described in Section 2.1) requires extra processing to be represented as hierarchical clusters. Here, we provide an algorithm to convert our JSON data into a series of progressively smaller clustering matrices (see Algorithm 1).

Essentially, we first traverse the outer voices of the JSON file, clustering notes of depth 0 into the closest note of higher depth to the left in the same voice. If that note does not exist, it defaults to the closest note of a higher depth to the right. For inner voices, if they do not describe hierarchical depth (all 0 depth), they are clustered 50%-50% between the nearest bass and soprano below and above or left to right, in that order. If the inner voice has

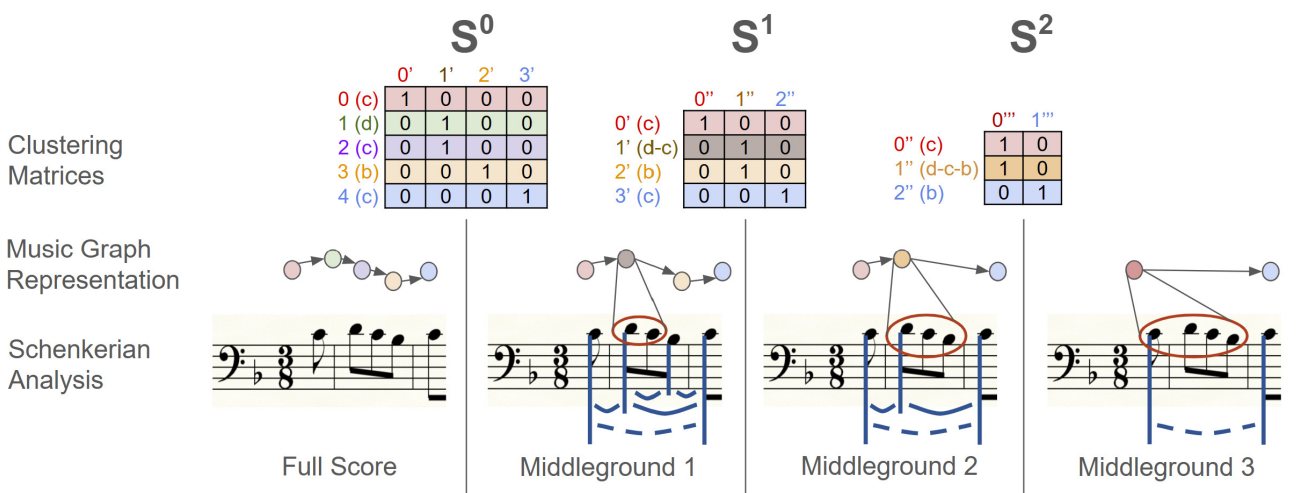


Figure 5: Visualization of Schenkerian analysis as a series of clustering matrices. The bottom row shows a simple score with Schenkerian annotation moving from all notes in the score to more abstracted versions of the score from left to right. The middle row visualizes the music as a graph. The top row shows the ground truth cluster matrices that relate one layer to the next; rows describe nodes before clustering, while columns describe nodes after clustering.

specified depth, it is treated similarly to the outer voices. All depths are then decremented and the process begins again for the next clustering matrix.

3.4 Implications of SchA Graph Clustering

The above formulation of SchA as a graph clustering problem facilitates more generalizable analysis. Whereas Kirilin’s MOP-based model focuses on a single melody as one theoretical voice, a fuller graph representation allows for greater flexibility via any number of theoretical voices. There are, however, several drawbacks with this new approach. Because the clustering works with the notes of the score, it is unclear how to handle cases where multiple theoretical voices converge on a single note. This issue may also be present when handling inner voices of unspecified depth. In our algorithm, we suggest splitting unspecified inner voices 50%-50% between the outer voices, but other approaches may also be reasonable.

Another advantage that the proposed graph clustering representation has over the MOP representation is its ability to cluster multiple notes into one in a single layer. This is particularly common when there are several repeated notes. In a MOP, repeated notes must be given detailed hierarchy, whereas a human expert would generally think of such repetitions as structurally redundant. There are also instances of prolongations that span more than one child, where having only one child would not properly reflect the music. For instance, if the melody over a C major tonic triad (CEG) quickly plays out the upper tetrachord of the scale, G-A-B-C, then the A and B are structurally equal; they both bridge the gap from G to C. On the other hand, allowing multiple children for every prolongation makes the search space for potential solutions orders of magnitude larger.

As the amount of labeled SchA data grows and computational power improves, there is great potential for learning complex relationships via machine learning that may

be unattainable in previous analyses. Deep learning has enjoyed considerable success on analyzing the Bach chorale dataset [39–41], thus we are optimistic that SchA can also be learned for broad datasets from different genres. The proposed dataset, notation software and graph representation provides a promising step towards this goal.

4. CONCLUSION

In this paper, we introduce the largest corpus of Schenkerian analyses in computer-readable format to date. This was largely made possible using our novel SchA notation software, which is natural, interpretable, and enables easy data collection and visualization. Finally, we describe and discuss a novel representation for SchA as a graph clustering problem that allows representation of any possible Schenkerian analysis, avoiding the limitations of MOPs. It is our hope that the growing amount of data and ease of its collection will enable broader research into SchA’s applications.

5. REFERENCES

- [1] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, “Simple and controllable music generation,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [2] P. Dhariwal, H. Jun, C. Payne, J. W. Kim, A. Radford, and I. Sutskever, “Jukebox: A generative model for music,” *arXiv preprint arXiv:2005.00341*, 2020.
- [3] Q. Huang, D. S. Park, T. Wang, T. I. Denk, A. Ly, N. Chen, Z. Zhang, Z. Zhang, J. Yu, C. Frank *et al.*, “Noise2music: Text-conditioned music generation with diffusion models,” *arXiv preprint arXiv:2302.03917*, 2023.

- [4] R. Castellon, C. Donahue, and P. Liang, “Codified audio language modeling learns useful representations for music information retrieval,” *arXiv preprint arXiv:2107.05677*, 2021.
- [5] M. Won, S. Oramas, O. Nieto, F. Gouyon, and X. Serra, “Multimodal metric learning for tag-based music retrieval,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 591–595.
- [6] P. B. Kirilin, “A data set for computational studies of schenkerian analysis.” in *ISMIR*, 2014, pp. 213–218.
- [7] B. Finane, “The humanist - murray perahia - steinway & sons.” [Online]. Available: <https://www.steinway.com/news/features/the-humanist-murray-perahia>
- [8] H. Schenker, *The art of performance*. Oxford University Press, 2000.
- [9] T. L. Jackson, “Heinrich schenker as composition teacher: The schenker-oppel exchange,” *Music Analysis*, vol. 20, no. 1, pp. 1–115, 2001.
- [10] D. F. Nobile, “A structural approach to the analysis of rock music,” Ph.D. dissertation, 2014, copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2023-03-03. [Online]. Available: <https://login.proxy.lib.duke.edu/login?url=https://www.proquest.com/dissertations-theses/structural-approach-analysis-rock-music/docview/1506971540/se-2>
- [11] J. Stock, “The application of schenkerian analysis to ethnomusicology: problems and possibilities,” *Music Analysis*, vol. 12, no. 2, pp. 215–240, 1993.
- [12] S. Larson, *Analyzing Jazz: A Schenkerian Approach*, ser. ACLS Humanities E-Book. Pendragon, 2009. [Online]. Available: <https://books.google.com/books?id=CmMJAQAAMAAJ>
- [13] A. Didier, “Form and tonal spectrum in 12-tone music: Approaches to analysis in schoenberg, walker, and webern,” Ph.D. dissertation, University of Oregon, 2022.
- [14] H. Fong, V. Kumar, and K. Sudhir, “A theory-based interpretable deep learning architecture for music emotion,” *Available at SSRN 4025386*, 2023.
- [15] J. Wu, C. Hu, Y. Wang, X. Hu, and J. Zhu, “A hierarchical recurrent neural network for symbolic melody generation,” *IEEE transactions on cybernetics*, vol. 50, no. 6, pp. 2749–2757, 2019.
- [16] S. Hahn, R. Zhu, S. Mak, C. Rudin, and Y. Jiang, “An interpretable, flexible, and interactive probabilistic framework for melody generation,” in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2023, pp. 4089–4099.
- [17] X. Zhang, J. Zhang, Y. Qiu, L. Wang, and J. Zhou, “Structure-enhanced pop music generation via harmony-aware learning,” in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 1204–1213.
- [18] W. N. Rothstein, “Phrase rhythm in tonal music,” (*No Title*), 1989.
- [19] F. Lerdahl and R. S. Jackendoff, *A Generative Theory of Tonal Music, reissue, with a new preface*. MIT press, 1996.
- [20] J. Hepokoski and W. Darcy, *Elements of sonata theory: Norms, types, and deformations in the late-eighteenth-century sonata*. Oxford University Press, 2006.
- [21] W. E. Caplin, *Analyzing classical form: An approach for the classroom*. Oxford University Press, USA, 2013.
- [22] H. Schenker, *Free Composition: Volume III of new musical theories and fantasies*. Pendragon Press, 2001, vol. 1.
- [23] A. C. Cadwallader, D. Gagné, and F. Samarotto, “Analysis of tonal music: a schenkerian approach,” (*No Title*), 1998.
- [24] C. Burkhardt, “Schenker’s” motivic parallelisms,” *Journal of Music Theory*, vol. 22, no. 2, pp. 145–175, 1978.
- [25] R. Gauldin, “Beethoven, tristan, and the beatles,” in *College Music Symposium*, vol. 30, no. 1. JSTOR, 1990, pp. 142–152.
- [26] M. Kassler, *Proving musical theorems I: The middleground of Heinrich Schenker’s theory of tonality*. Basser Department of Computer Science, School of Physics, University of Sydney, 1975, no. 103.
- [27] R. E. Frankel, S. J. Rosenschein, and S. W. Smoliar, “A lisp-based system for the study of schenkerian analysis,” *Computers and the Humanities*, pp. 21–32, 1976.
- [28] S. W. Smoliar, “A computer aid for schenkerian analysis,” in *Proceedings of the 1979 annual conference*, 1979, pp. 110–115.
- [29] P. Mavromatis and M. Brown, “Parsing context-free grammars for music: A computational model of schenkerian analysis,” in *Proceedings of the 8th International Conference on Music Perception & Cognition*, 2004, pp. 414–415.
- [30] É. Gilbert and D. Conklin, “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*. Citeseer, 2007, pp. 83–94.
- [31] A. Marsden, “Schenkerian analysis by computer: A proof of concept,” *Journal of New Music Research*, vol. 39, no. 3, pp. 269–289, 2010.

- [32] P. B. Kirlin, “A probabilistic model of hierarchical music analysis,” PhD thesis, University of Massachusetts Amherst, Amherst, MA, February 2014, available at <https://www.cs.rhodes.edu/~kirlin/p/diss.html>.
- [33] J. D. Yust, *Formal models of prolongation*. ProQuest, 2006.
- [34] A. Forte and S. E. Gilbert, “Instructor’s manual for introduction to schenkerian analysis,” (*No Title*), 1982.
- [35] A. Forte and S. Gilbert, “Introduction to schenkerian analysis: instructor’s manual,” 1982.
- [36] T. Pankhurst, *SchenkerGUIDE: a brief handbook and website for Schenkerian analysis*. Routledge, 2008.
- [37] P. Kirlin and D. Jensen, “Learning to uncover deep musical structure,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.
- [38] D. Jeong, T. Kwon, Y. Kim, and J. Nam, “Graph neural network for music score data and modeling expressive piano performance,” in *International conference on machine learning*. PMLR, 2019, pp. 3060–3070.
- [39] G. Hadjeres, F. Pachet, and F. Nielsen, “Deepbach: a steerable model for bach chorales generation,” in *International conference on machine learning*. PMLR, 2017, pp. 1362–1371.
- [40] C.-Z. A. Huang, A. Vaswani, J. Uszkoreit, N. Shazeer, C. Hawthorne, A. M. Dai, M. D. Hoffman, and D. Eck, “An improved relative self-attention mechanism for transformer with application to music generation,” *arXiv preprint arXiv:1809.04281*, vol. 2, 2018.
- [41] F. T. Liang, M. Gotham, M. Johnson, and J. Shotton, “Automatic stylistic composition of bach chorales with deep lstm.” in *ISMIR*, 2017, pp. 449–456.

6. ETHICS STATEMENT

This particular work has no direct negative ethical implications.

We acknowledge that Heinrich Schenker (the inventor of Schenkerian analysis) was racist and nationalist. His sociopolitical views are not condoned by authors of the present work. As originally designed, his style of analysis did not extend far beyond German composers of the common practice era. As we address in Section 1, we do not see Schenkerian analysis as a static analysis defined by Schenker; rather, we see it as a growing and developing set of tools that may be applied to various musical genres, detached from Schenker’s personal views.