PHRASES IN PERFORMANCE AND PHRASES IN THE SCORE

Romane Clerc[1] **Octavio Profeta**[2] **Cindy Tang**[1] **Shu Yang**[1]

[1] Digital Humanities, EPFL

[2] Robotics, EPFL

ABSTRACT

Phrasing is a fundamental concept in both speech and music, playing a crucial role in structuring auditory streams and organizing auditory information. This study aims to identify the characteristics that define musical phrase boundaries in terms of performance and score attributes. We developed three different phrase segmentation algorithms based on velocity and cadence. Our findings reveal that velocity-based models are particularly sensitive to dynamic nuances, while the cadence-based model is more responsive to significant harmonic shifts, offering complementary insights into musical phrasing. It would be our possible future work to incorporate both performance attributes and symbolic scores to make our algorithm more applicable.

1. INTRODUCTION

Speech and music play pivotal roles in human communication. They serve as mediums for conveying meaning, emotion, and expression. While speech serves as a primary vehicle for linguistic communication, music, with its intricate interplay of melody, harmony, rhythm, and timbre, offers a unique avenue for emotional and artistic expression.

At the heart of both speech and music lies the concept of phrasing, a fundamental aspect that structures auditory streams and aids in the organization of auditory information [1, 2]. Phrasing delineates segments within a continuous auditory stream, providing listeners with perceptual cues to discern meaningful units of information [2, 3].

In speech, a phrase represents a low-level component within the hierarchical structure, typically comprising small groups of words that collectively form larger meaning. Acoustic cues such as changes in intensity, rhythmic patterns, and pitch contour serve as primary determinants for discerning boundaries between segments in speech, facilitating comprehension and interpretation in communication between humans [4, 5].

In music, contrasts in pitch range, changes in melodic contour, tonal stress, and rhythmic patterns contribute to the delineation of musical phrases, guiding listeners through the expressive journey crafted by the composer and performer [1, 2]. Notably, a research indicates that even infants as young as six months exhibit preferences for musical segments delineated by pitch variations and rhythmic structures, underscoring the early emergence of phrasing perceptual mechanisms in human development [6].

Others studies have been made to develop computational models aimed at detecting phrase boundaries in music, based on features such as note onsets [7] or tempo [8]. These endeavors reflect an interest in understanding the underlying mechanisms of phrasing perception and segmentation, bridging the gap between cognitive science, musicology, and computational analysis.

Drawing from the insights gained in these previous studies, our study embarks on a parallel journey. Our primary objective is to engage in a close listening session of Mozart's *Piano Sonata n°8 I* piece, aiming to gain a comprehensive understanding of what constitutes a musical phrase and the defining characteristics of phrase boundaries in terms of performance attributes and MIDI score analysis. Subsequently, we undertake the manual annotation of phrases boundaries on the symbolic score. We then develop a phrase segmentation algorithm based on our previous observations. Finally, we evaluate our model by comparing its segmentation results with the manual annotations and conducting statistical analyses across multiple musical pieces.

2. DATA

The data utilized comes from the Aligned Scores and Performances (ASAP) dataset [1], which comprises 236 musical scores with 1067 performances from 15 composers, including our piece of interest, Mozart's *Piano Sonata n°8 I*.

[1] https://github.com/fosfrancesco/asap-dataset

3. WHAT IS A PHRASE?

Phrases in music can emerge from different context clues; from a listening experience, we can take a more specific look into what makes a phrase.

3.1 Dynamics

Phrases seem to have well defined dynamic for themselves, however each phrase also seem to have a dynamic of their own; for example, by looking at the velocity curve in the unperformed MIDI score of Mozart's *Piano Sonata n°8.1*, we can see that the velocity of the first phrase is of approximately 96, while the velocity of the second phrase is at approximately 64. While it makes sense for unperformed scores to have constant velocity for phrases, we can interpolate that performed version of the same phrases might have variation in their own velocity, but we can assume that performances will tend to keep the same variation throughout the piece.

3.2 Accents

Accentuation of certain note helps to mark beginning and ending of phrases; when hearing a performance, start of phrases are clearly accentuated, meaning the performer voluntarily plays the notes harder than normal, while end of phrases tend to be de-accentuated, as to leave space for what is to come - in general, a new phrase.

3.3 Cadence

As seen during the course, cadences are the ending of phrases, which in term create a sense of partial or full resolution the phrase. Just as points, commas or question mark the end of grammatical phrases, different cadences mark the end of different musical phrases.

4. CLOSE LISTENING OF THE PIECE

With this report is attached an annotated PDF of the score for the unperformed version of the sonata. We annotated the score with the phrase boundaries that we noticed in the sonata. Phrase boundaries recognized with both listening and reading the score are marked in green while those recognized only by listening are marked in blue, and the one recognized only by reading the score are marked in yellow. Additionally to the boundaries, we tried to differentiate the cadences that marked the end of the phrases, with the letters IAC for Imperfect Authentic Cadence, HC for Half Cadence, and PAC for Perfect Authentic Cadence. We also delimited the structure of almost every phrase of the sonata with the red annotations BI for Basic Idea, BI' for the consequent, that is close to the Basic Idea, and CI for for the Complementary Idea.

4.1 Hearing versus understanding

Only a few boundaries that were not noticed when reading the score but seemed obvious when listening to the performance. These were in bars 40, 62, 66 and 129. The first boundary was missed because of the absence of a chord and the continuous use of sixteenth notes. We probably missed the next two ones because of the key change, from C to Am, which is easily perceptible in a performed piece but is harder to spot when reading a score. We will later argue that this key change might, in the contrary, have led us into thinking there were different phrases. Finally, it is possible that the limit of bar 129 has escaped us because of its proximity to the end of the score. In conclusion, most of the boundaries were noticed by reading the score, as the easiest way to do so is to imagine the music being played, which, in the end, remains pretty close to listening to a performance. There were still two boundaries, that we had not spotted at first, but which we noticed as we studied the structure of each phrase, in bars 70 and 74.

4.2 Structure of a phrase

Having separated each phrase, their structure became quite obvious. Each phrase seems to be composed by a basic idea, then the same basic idea but with a little variation, and finally a continuation leading to the cadence. The structure stays the same during all the piece, while the lengths of each part varies. On a short phrase, each basic idea can last from one bar (e.g. the phrase from bar 101 to bar 103) to six bars (i.g. the phrase from bar 23 to bar 34). This structure is constant through the piece. Listening to the performances, we noticed from bar 58 to bar 79, phrases boundaries that we had not spotted when reading the score. They actually correspond to different ideas in the same phrase, but as each new idea involves a switch in key, we mistakenly thought they were separate phrases. There remains a phrase for which we could not find a structure, from bar 104 to bar 115, where there are many changes of melody, which does not seem to follow a three-part pattern.

In performances, the pianists usually repeat both basic ideas with the same intensity, even if they lowered the sound at the end of the first idea, the second one will begin with the same intensity the first one started. However, when the repetition of the basic idea differs from the original one, the performer will accentuate it, playing it louder. If the idea is longer, as for the phrase starting in the 23rd bar, its repetition will be played quieter, as to keep the listener engaged. When the complementary idea differs a lot from the basic ideas, it will be played louder, as in the bars 54 of 73.

The cadences do not seem to follow any particular pattern, but we may sometimes have got them wrong. The final chord of the piece is a Am chord, but with a C as the highest note. This made us think that the piece ends with an Imperfect Authentic Cadence, which is quite unusual. Given that the piece we're analyzing is only the first movement - Allegro Maestoso - and there are still two more to follow, it's not absurd to it with an Imperfect Authentic Cadence, in order to keep the listener intrigued and eager to hear the rest of the sonata.

5. METHODS

We initially developed a velocity-based phrase segmentation model, `Model_P_velocity`. To enhance its performance, we then developed `Model_P_velocity_with_SSM`, an improved version that utilizes a Self-Similarity Matrix (SSM) to capture velocity changes instead of employing a fixed threshold as in the base model. Additionally, we developed a third model, `Model_P_cadence`, which is based on cadence features.

The main toolkit employed is music21 [9], a Python-based toolkit designed for computer-aided musicology, to extract essential features such as notes, chords, pitches or velocity from the dataset.

All our code can be accessed on GitHub [2].

5.1 Velocity-based phrase segmentation algorithm

Drawing from our prior investigations, we opt to develop a phrase segmentation algorithm centered on velocity information. We opt for velocity as it lends itself to straightforward computational extraction from MIDI files, and our preliminary observations suggest promising outcomes. Thus, we introduce the function `Model_P_velocity`, that takes a MIDI score as input and generates a list of measure numbers corresponding to detected phrase boundaries. The algorithm traverses through each note and chord, detects where there is a significant change in velocity between two consecutive elements and returns the corresponding measure number.

In MIDI files, velocity values range from 0 to 127, with the minimum detectable change in velocity observed to be 10 during our examination of the unperformed rendition of Mozart's *Piano Sonata n°8 I*. Thus, we set the threshold for detecting phrase boundaries at 10. Any disparity in velocity exceeding this threshold between two consecutive notes or chords is considered a significant change indicative of a phrase boundary.

5.2 Velocity-based phrase segmentation algorithm using Self-Similarity Matrix (SSM)

We have noticed that even within the same pieces, different performers may interpret the notes uniquely, leading to variations in the number of measures for the same piece. Additionally, variations in velocity may arise based on individual performance styles, which could mean that our model design distinguishing boundaries with a velocity difference of fixed 10 might overlook the nuances between different performers' renditions of the piece to some degree. Therefore, in considering our model, we opted to use a Self-Similarity Matrix (SSM) to captures changes in note velocities. By analyzing how each note's velocity compares to others, the model detects significant shifts in musical expression, hypothesized as phrase transitions. This method offers a nuanced approach that adapts to individual performance styles. We believe it may mitigate structural differences in music caused by different performers or versions. This method does not rely directly on the absolute length of measures but focuses on the relationships and dynamic changes between elements within the music, thereby better capturing unique treatments and expressions in performance. Thus, we hope SSM can improve our model's precision. Initially, we designed our matrix using cosine similarity to compute the self-similarity matrix; however, we found it ineffective at detecting boundaries. Therefore, we switched to using Euclidean distance to design our SSM.

5.3 Cadence-based phrase segmentation algorithm

Finally, we have also explored the potential of using cadence information for our phrase segmentation algorithm. To this end, we have developed the `Model_P_cadence` function, which, similar to `Model_P_velocity`, accepts a MIDI score as input and outputs a list of measure numbers indicating identified phrase boundaries. This algorithm systematically iterates through each chord, identifying instances where two consecutive chords consist of pitches of the EGB and AEC sets.

6. RESULTS

6.1 Comparison of model-predicted phrases with self-marked ones

First, we compare the phrases predicted by our models with those manually annotated by us. We evaluate our models' accuracy both manually and using precision, recall, and F1 score metrics.

6.1.1 Model_P_velocity

```
Model_P_velocity Boundaries = {1, 5,
6, 9, 15, 16, 18, 20, 22, 23, 54, 55,
56, 62, 66, 84, 85, 88, 99, 101, 103,
104}

Self-marked Boundaries = {5, 9, 16,
22, 35, 40, 49, 58, 62, 66, 70, 80,
84, 88, 94, 97, 101, 103, 116, 129,
133}
```

We ran our model to the unperformed version of Mozart's *Piano Sonata n°8 I* and compared the results with the ones annotated by hand. We then identified common boundaries, boundaries missed by our model, and boundaries uniquely detected by our model. Based on these data, we calculated the precision, recall, and F1 score and got the result as below: precision: 0.45, Recall: 0.56 and F1 Score: 0.5, our model demonstrates moderate effectiveness in identifying correct musical phrase boundaries but still shows room for improvement in reducing false positives and capturing missed boundaries.

[2] `https://github.com/cindytangch/phrases-in-music-performance`

6.1.2 Model_P_velocity_with_SSM

By using `Model_P_velocity_with_SSM`, we got the results comparing to our self-marked one as below: Precision: 0.48 , Recall: 0.61 and F1 Score: 0.54 on the unperformed version of the Mozart's *Piano Sonata n°8 I* piece. This improvement may suggest the adjustments made have positively impacted our model's ability to comprehensively detect boundaries, which indicating a more robust model performance overall.

6.2 Statistical Analysis

In order to analyze the performance of our models across larger databases, thereby understanding the characteristics of our model, we have chosen the entire dataset of Mozart's *Piano Sonatas* as the basis for our next phase of statistical analysis. Firstly, by applying a more intuitive visualization approach, we plotted the phrase boundaries of different versions (both performed and unperformed) of Mozart's Piano Sonatas series. By observing, we can gain some general insights into how the three models capture phrase information.

6.2.1 Model_P_velocity vs. Model_P_cadence

By comparing the distributions of detected phrase boundaries from `Model_P_velocity` in Figure 1 and `Model_P_cadence` in Figure 2, we observed that the boundaries obtained using `Model_P_velocity` are relatively denser than those detected by `Model_P_cadence`. This discrepancy might be due to the velocity-based model's heightened sensitivity to minor dynamic changes, whereas the cadence-based model likely identifies boundaries only at significant harmonic or rhythmic changes. Therefore, we believe `Model_P_velocity` would be more sensitive to subtle performance variations, which would make it more effective in discerning the distinct characteristics of different performers. However at the same time, it would be also more susceptible to variations or even disturbances caused by the performers' state during a performance. Regarding changes based on cadence, we can see that the phrases are noticeably sparser, but these cadence-based changes are often more decisive within movements, which aids in our understanding of the overall structure of the music. However, as seen in piece 12-2, the `Model_P_cadence` has almost failed to detect any boundaries, indicating the model heavily relies on the composition of the music, if the composer does not employ such harmonic structures, then the corresponding boundaries are often undetectable, suggesting that this model lacks adaptability based on composition to some extent.

6.2.2 Model_P_velocity vs. Model_P_velocity_with_SSM

From Figure 1 and Figure 3, we had a comparison of `Model_P_velocity` with `Model_P_velocity_with_SSM`. A significant change is apparent in the piece 12-2, where the boundaries determined by the `Model_P_velocity_with_SSM` are noticeably denser. This has drawn our attention. We have noted that in the two performed versions of 12-2, the boundaries of individual phrases within the music are quite short, indicating shorter phrase lengths. The presence of multiple dynamic changes within a short duration seems to facilitate the `Model_P_velocity_with_SSM` algorithm in identifying more local similarities and differences during the analysis of these segments. Consequently, for a piece like 12-2, which is perhaps faster and involves more changes, the boundaries tend to be denser. We assumed that smaller windows often capture more changes, thus resulting in denser boundary markings. What is more, from observing these two graphs, we can also see that with the other pieces of music, the `Model_P_velocity_with_SSM` usually delineates slightly denser boundaries, and the distributions of boundaries of both plots show a similar trend. Therefore, to some extent, `Model_P_velocity_with_SSM` may enhance the precision of boundary division to a certain extent, but the computational cost it incurs also needs to be considered.

6.2.3 In-depth quantitative assessment

Furthermore, to conduct a deeper observation of the performance of the three models, we have also created charts for average phrase count, average phrase length and other metrics, which can be seen in Figure 4, 5, 6. It is observed that `Model_P_velocity` has a higher standard deviation compared to `Model_P_velocity_with_SSM`. This suggests that the `Model_P_velocity_with_SSM` model produces more consistent phrase lengths, indicating that it maintains a relatively stable recognition standard across different parts of the music, likely resisting minor variations well. Additionally, while `Model_P_velocity` and `Model_P_velocity_with_SSM` show similar trends in average length and average count, observations from the left y-axis of the chart confirm our previous findings that `Model_P_velocity_with_SSM` has a higher average phrase count than `Model_P_velocity`, which again reflects that SSM method may identify more phrase boundaries by detecting subtle similarities and differences in complex or dynamic segments. `Model_P_cadence`, on the other hand, shows significantly fewer phrases and notably longer phrase lengths, suggesting that this method of boundary detection may not be very adaptive and may perform poorly in compositions without clear corresponding chord changes, yet its boundaries are often more impactful. Another consideration is that in pieces with strict structures and regular rhythms, the difference between `Model_P_velocity` and `Model_P_velocity_with_SSM` may not be significant; thus, considering computational costs, the former might be a good choice. However, for music that tends towards high improvisation and broad changes, such as jazz and electronic dance music (EDM), using the latter may be a better option.

6.3 Qualitative assessment based on one specific piece

From our before analysis, we observed that the two velocity-based algorithms identified very dense phrase boundaries in piece 12-2. Therefore, we conducted a detailed listening and analysis to determine whether these computationally detected results align with our perceptual experience. However, after listening, we found that not all of the phrase boundaries identified by the models resonated as expected. This insight on the excessive segmentation of the piece may lead to reflection for the next steps in optimizing our models in the future.

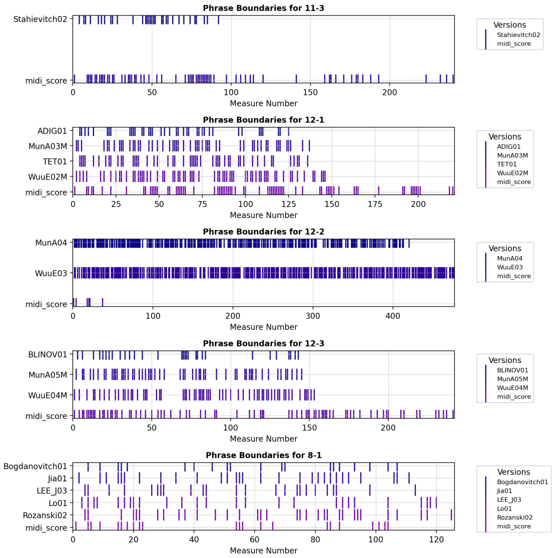

Figure 1. Distribution of detected phrase boundaries by the `Model_P_velocity` algorithm on Mozart's *Piano Sonatas*. 'midi_score' is referred to the unperformed version, while the other ones are refereed to different performed versions of the piece.

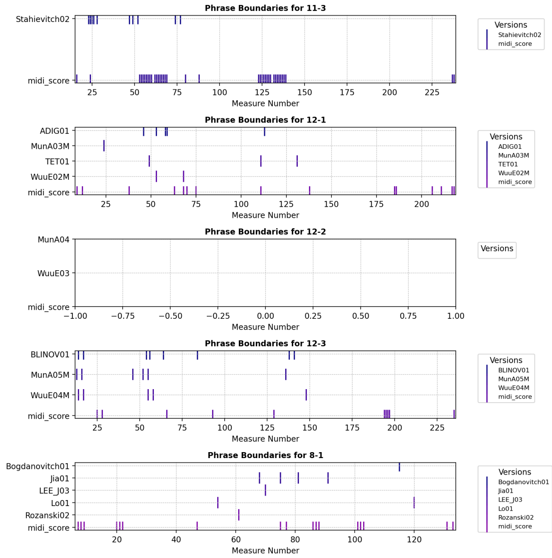

Figure 2. Distribution of detected phrase boundaries by the `Model_P_cadence` algorithm on Mozart's *Piano Sonatas*. 'midi_score' is referred to the unperformed version, while the other ones are refereed to different performed versions of the piece.

7. DISCUSSION

Our models exhibits certain limitations. Firstly, it solely identifies the measure at which a phrase boundary occurs, without distinguishing whether the boundary lies at the beginning, middle, or end of the measure. This design choice comes from our observation that significant changes in velocity often occur within a consecutive set of notes or chords surrounding phrase boundaries. By focusing solely on note iterations and velocity changes, the algorithm could potentially generate multiple signals for one phrase boundary, contrary to our expectations. Hence, we opt for a less precise approach that only determines the measures where the phrase boundaries occur instead of its exact position in the measure.

To evaluate our models' accuracy, we only compared the phrase boundaries detected by our models with the ones annotated by hand for only one piece, which may not be convincing enough to ensure our model's changes. Going forward, we hope to validate and refine our model's accuracy with a broader dataset, it would be great we could have more annotated data.

Based on our recent work, we have identified a potential issue with our velocity-based algorithms, including both `Model_P_velocity_with_SSM` and `Model_P_velocity`. These algorithms can sometimes be overly sensitive to phrases. For instance, in piece 12-2, the phrases are densely packed, which might not accurately reflect the complex psychological perceptions involved. As the Prof. Martin Rohrmeier pointed out in class, phrases are, to some extent, a functional cognitive process and are not perceived identically by everyone. We suggest that sometimes a psychological state may not directly correlate with computational expressiveness. In other words, a pattern recognized computationally may not truly represent how the human brain processes it. Overly dense phrases, as identified computationally, could potentially lead to cognitive overload, which might not be cognitively feasible.

In future analyses, one possible approach is to combine our velocity-based algorithm with a cadence-based function to achieve a deeper segmentation of the tracks. This would ensure that we identify the more crucial phrase boundaries while effectively eliminating some that may be unnecessary or redundant. Furthermore, to compensate for the shortcomings of the cadence function, we could try to identify other significant chord change pattern based on the overall structure of the specific music and conduct a deeper analysis based on changes in cadence.

Additionally, we could enhance our SSM method by incorporating other factors which may significantly influ-

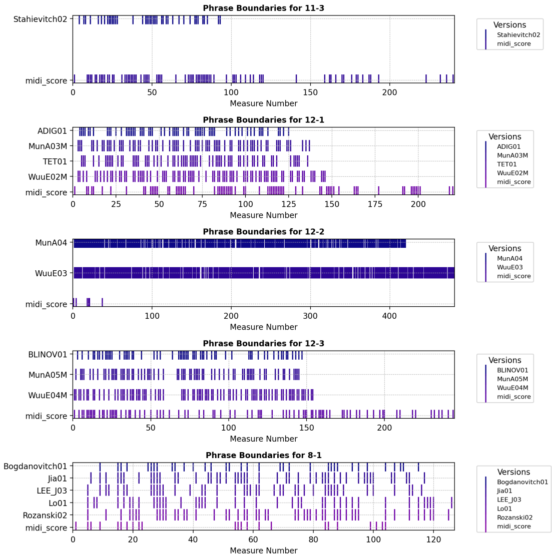

Figure 3. Distribution of detected phrase boundaries by the `Model_P_velocity_with_SSM` algorithm on Mozart's *Piano Sonatas*. 'midi_score' is referred to the unperformed version, while the other ones are refereed to different performed versions of the piece.

ence human perception, thereby enriching our analysis with more elements and being more thorough.

8. CONCLUSION

Our investigation into phrase segmentation algorithms has underscored the complexity of identifying precise phrase boundaries in music. Through the development of three distinct models based on velocity and cadence, we have gleaned that different characteristics of phrases yield varying results. Velocity-based models proved sensitive to dynamic nuances, whereas cadence-based models were more attuned to significant harmonic shifts, each offering valuable, complementary insights into musical phrasing.

This project also deepened our appreciation of the intricate parallels between music and speech. Both forms of communication are profoundly rooted in personal interpretation, influenced by culture, experiences, and individual perception. Music, like speech, exists within the delicate interplay between reality and the mind, serving as a medium for conveying ideas. It is constructed with the same foundational elements as language, reflecting its role as a fundamental form of human expression.

Looking forward, a promising avenue for future work lies in integrating both performance attributes and symbolic scores to enhance the applicability of our phrase segmentation algorithms. By combining these elements, we aim to develop more robust models that can better capture the multifaceted nature of musical phrasing. Our research thus far has provided a solid foundation, and we anticipate that future advancements will further elucidate the complexities of musical phrase boundaries.

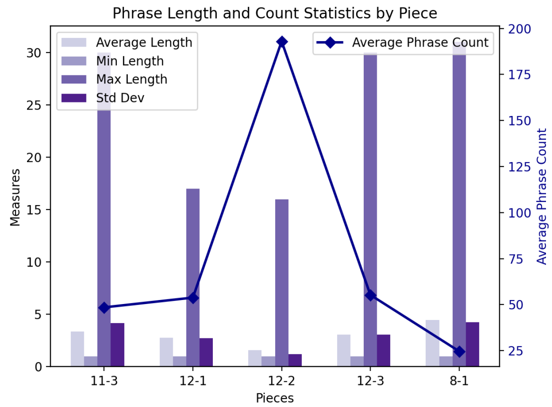

Figure 4. Statistical analysis of average phrase length, average phrase count, and other metrics using the `Model_P_velocity` algorithm.

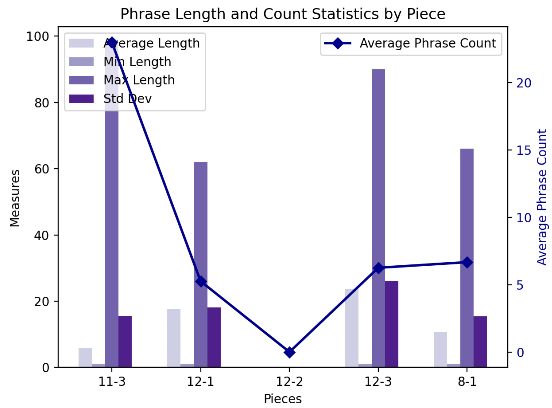

Figure 5. Statistical analysis of average phrase length, average phrase count, and other metrics using the `Model_P_cadence` algorithm.

9. AUTHOR CONTRIBUTIONS

Initially, tasks were divided among the team members: O. Profeta conducted close listening and analysis of the performed piece, R. Clerc analysed and annotated phrases by hand, C. Tang reviewed the literature and implemented the phrase segmentation algorithm, and S. Yang improved, analyzed and evaluated the model. Each member starts by analyzing and drafting their respective sections of the report. Subsequently, all team members actively engaged in reviewing, providing feedback, and refining each section. This collaborative approach enabled each section to benefit from the diverse expertise of each team member.

10. REFERENCES

[1] K. N. Olsen, R. T. Dean, and Y. Leung, "What constitutes a phrase in sound-based music? a mixed-methods investigation of perception and acoustics," *PloS one*, vol. 11, no. 12, p. e0167643, 2016.

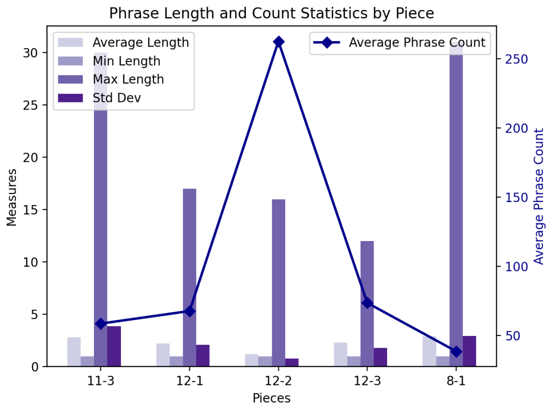

Figure 6. Statistical analysis of average phrase length, average phrase count, and other metrics using the `Model_P_velocity_with_SSM` algorithm.

[2] T. R. Knösche, C. Neuhaus, J. Haueisen, K. Alter, B. Maess, O. W. Witte, and A. D. Friederici, "Perception of phrase structure in music," *Human Brain Mapping*, vol. 24, no. 4, pp. 259–273, 2005.

[3] A. S. Bregman, *Auditory scene analysis: The perceptual organization of sound*. MIT press, 1994.

[4] A. Cutler, "Exploiting prosodic probabilities in speech segmentation," *G. T. M. Altmann (Ed.), Cognitive models of speech processing: Psycholinguistic and computational perspectives*, pp. 105–121, 1991.

[5] L. A. Streeter, "Acoustic determinants of phrase boundary perception," *The Journal of the Acoustical Society of America*, vol. 64, no. 6, pp. 1582–1592, 1978.

[6] C. L. Krumhansl and P. W. Jusczyk, "Infants' perception of phrase structure in music," *Psychological science*, vol. 1, no. 1, pp. 70–73, 1990.

[7] Z. Cenkerová, M. Hartmann, and P. Toiviainen, "Crossing phrase boundaries in music," in *Proceedings of the Sound and Music Computing Conferences*, 2018.

[8] C.-H. Chuan and E. Chew, "A dynamic programming approach to the extraction of phrase boundaries from tempo variations in expressive performances." in *ISMIR*, 2007, pp. 305–308.

[9] M. S. Cuthbert and C. Ariza, "music21: A toolkit for computer-aided musicology and symbolic music data," *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR 2010)*, pp. 637–642, 2010.

The writing of this work was assisted by OpenAI's ChatGPT (version 3.5) for text reformulations and grammar checks. Retrieved from `https://www.openai.com/chatgpt`.