Generating a "Human-Like" Slangpolska Piece

Mariella Daghfal, Kaede Johnson, Shayan Khajehnouri, & Tahel Singer
EPFL Course DH-501

I. Introduction

We seek to computationally generate a 'human-like' score that emulates the rhythmic, melodic, and structural schemata underpinning a corpus of Swedish Slangpolska music. Slangpolska refers to a type of folk music that is usually played to accompany a dance of the same name, danced by grouping four dance steps into three beats, which is reflected in the meter of the music.

We conceptualize pitch as a means of communicating melody, rhythm as a pattern of event onsets, durations, and rests in reference to an underlying metrical structure, and form as the repetition and overarching organization of musical material. Appropriate patterns in these components enable humans to follow the traditional dance steps of Slangpolska music, underscoring the importance of "human-like" score generation for our selected corpus.

After designing a model that generates scores at random and a model that generates scores using key-, duration-, onset-, and pitch-informed bigrams, we compare results from the two by way of disparities in pitch sequences, chronotonic chains, and FastText-trained measure embeddings. We close with a few limitations of our model and suggestions for future work.

II. Representing a Musical Score

In an abstract sense, we conceptualize score as the tone, interval, duration, and structural personality of a series of musical events which consistently respect the ¾ meter. These defining characteristics motivate our choice of generative model and evaluation strategy.

We motivate our emphasis on these characteristics in the subsections below.

A. Pre-Processing

We begin by removing 15 scores that were polyphonic or lacked a consistent ¾ meter and 8 scores containing less than 8 measures to isolate consistent training data of appropriate length. Exploratory analysis using Python's music21 package [1] revealed that less than 10% of our corpus' scores contain a note with an articulation and just 1% of our corpus' notes are articulated, leading us to ignore articulations in our analysis. Additionally, because less than 1% of all events in the first 8 measures of the remaining scores were chords, we exclude 64 scores with early chord presence from our score comparisons and disregard chords in our generative model (those 64 scores with a chord in the first 8 measures are retained for model development and FastText embedding training). Our corpus for model development amounted to 576 scores, while our final Slangpolska corpus for distance calculations amounted to 512 scores.

B. Pitch

In targeting melody, we emphasize pitch expectancy. We believe pitch, the musical element which enables the communication of melody, applies to our corpus through the interplay of tonal proximity, interval direction, and "patterns of expectancy". All such qualities are captured in our Generative model - albeit with severely limited memory. The exclusion of key variance in our representation is a result of a desire for a denser training distribution.

C. Rhythm

There is no score without rhythm; there are no notes without places to put them. We isolate onsets, durations, rests, and their positional mapping to underlying meter as the core tenets of rhythm. Duration patterns especially are key to our framework. Though there are other important aspects of rhythm, notably syncopation and rubato, we do not consider them relevant to a traditional, (in this case) unperformed corpus.

D. Form

We see the repetition of musical motifs and the differing character of a score's beginning, middle, and end as form's tenets. In pursuit of these tenets, we emphasize measure-level control with our generative model. Because of the recursive nature of score's subdivisions, it is reasonable to model ultra-long-term dependencies in score generation; though we target form functions solely within the eight-measure span, our strategy for structural control can be readily extended to longer pieces.

III. Models

We developed two models in this assignment: a Random model and a Generative model.

The Random model builds measures by one 'event' (note or rest) at a time. It first selects among the dotted-half, half, quarter, eighth, or sixteenth durations at random. It then decides, again at random, whether this duration will apply to a note or a rest. Finally, if the event to be added is a note, the model randomly selects one of twelve pitches from the 12 equal temperament scale. Combining these three decisions, it adds the selected event to an empty measure with a ¾ time signature. It then removes any durations from the aforementioned set that would extend the measure beyond three beats (for example, if the Random model's first choice is the quarter note, it may no longer choose the dotted-half note

or rest). The Random model then proceeds to select a second duration at random, to decide if this duration will apply to a note or rest at random, to select a pitch at random (in the case of a note), to add this second event to the measure, and to filter the possible set of durations from its latest onset location. This process continues until the measure has been filled with events that total exactly 3 beats.

We built 500 Random scores by using the Random model to create and distribute 4000 measures into sets of eight. These 500 random 8-measure sequences act as the baseline for our Generative model.

By contrast, our Generative model, a Bigram model, builds scores based on a pattern of forms that we provide in the structure of a period. Our generative model works in two steps: In the first step, our Bigram model architecture is modeled after traditional natural language processing application. To begin a new score, the Generative model is fed a 'START' trio, while the onset, duration, and pitch of the first note of the first measure is selected according to the probability distribution of all the onset-duration-pitch trios found in score-starting notes in our corpus. The onset, duration, and pitch of the second note is then chosen according to the probability distribution of onset-duration-pitch trios that followed in the corpus the onset-duration-pitch trio just added to our generated measure. As with the Random model, however, selected durations must adhere to the ¾ meter. This process of looking backward one step for a probability distribution at the current onset persists even across the meter threshold. In the second step, the generative model groups eight measures generated in the first step in the form of a period structure ABAB'. This form is again subdivided into smaller patterns. For example, A is subdivided into two measures: a and a', where a' is the transposed version of a by a specified interval. B is also subdivided into b and c, and B' is subdivided into b and c', c' being the transposed version of c by a specific interval. The eight generated measures therefore follow the form of "aa'bcaa'bc'" To do the transposition, we created a function that transposes a measure to the specified number of intervals. Based on a list of twelve semitones, the function takes the index of the pitch of each note in the measure and adds the number of semitones it needs to transpose it to. It then calculates the remainder of the division of the obtained number by 12, before returning the transposed measure.

After using our Generative model to create 500 8-bar sequences, we calculated pairwise distances between our Random, Generated, and Actual scores according to the methods described in the next section.

IV. Evaluation Metrics

While our reported definition of score incorporates observable phenomena, we also believe a large part of experiencing music is purely mental. Accordingly, we sought distance metrics which involved our corpus' quantifiable aspects while also respecting human judgment. Toussant et al. [2] and Beltran et al. [3] find transformation methods for measuring rhythm distance - broadly, those which measure how much effort it takes to mutate one rhythm into another - superior at matching human judgment than features-based methods, which compare the presence of pre-defined rhythmic qualities between two scores. Additionally, Kelly finds transformation-based distance metrics broadly successful at comparing melodies relative to a human-judged 'ground truth' [4]. We therefore employ transformation-based methods in the evaluation of our model's pitch and rhythm distances. Meanwhile, inspired by Jhamtani and Berg-Kirkpatrick's success in matching human perception of self-repetition through a generative model based on measure embeddings [5], we pursue an distance metric based on measure embeddings when quantifying differences in form.

Our distance metric for **pitch** amounts to Levenshtein edit distance between pitch sequences with learned edit operation weights. We selected edit distance due its endorsement from Kelly [4] and use by Uitdenbogerd with pitch-only sequences [6]. First, we strip a sequence of measures into a consecutive string of letters representing pitch. For example, measures 1 and 2 of *Ninas slangpolska* (Figure **??**) are represented by the string "BGEBFGE", while measures 4 and 5 of *Dahl polska efter Ola Olsson* (Figure 1) are represented by the string "FGAABBCC". Note that rests do not affect these strings (except insofar as they take up space in the measure that could be used by pitched objects). Note further that we ignore key; all notes with pitch A are represented by the letter A. Flats are represented by a lowercase letter, and sharps are always mapped to their flat counterpart during string creation.

Under standard Levenshtein edit distance, the distance between "BGEBFGE" and "FGAABBCC" amounts to the smallest number of character insertions, deletions, and substitutions to convert the former string into the latter (in this case the distance is 6 - substitute the first B with F, insert an A after the first G, substitute the first E with A, substitute the original F with B, substitute the second G with C, and substitute the second E with C). Our metric is slightly different, as its weights are not uniformly 1. Instead, our weights are learned from our corpus - as is the case for Habrard et al [7] and Ristad & Yianilos [8].

To re-weight edit operations, we first calculate the distribution of all edit operations used to transform each score from our Slangpolska corpus into every other score in the Slangpolska corpus. We then take the log value of each edit operation's share of this aggregate distribution and divide each resulting value by the maximum from the resulting set (this maximum is, in other words, the log value of the most common edit operation's share of the original distribution). The resultant weights range between 1 (the most common edit operation's weight) and about 2.42 (the least common edit operation's weight). Edit operations not seen in the Slangpolska-to-Slangpolska Levenshtein calculations are automatically assigned the highest edit cost. Finally, symmetric operations with minor differences are equated to their average to preserve symmetry in the re-weighted function. The purpose of this re-weighting procedure is to punish frequent use of edit operations that are abnormal according to the pitch behavior of our Slangpolska corpus while keeping maximum edit costs

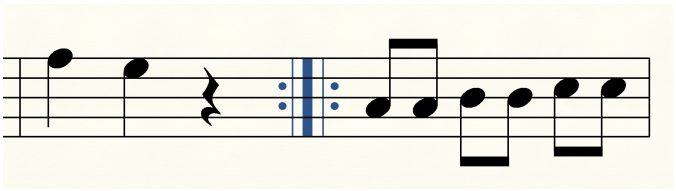

Fig. 1. Measures 4 and 5 of *Dahl polska efter Ola Olsson* as score

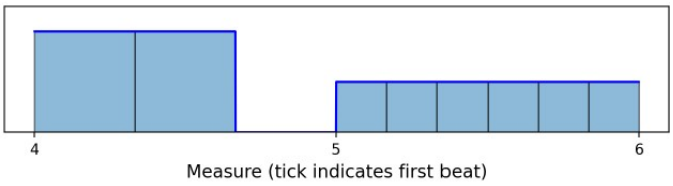

Fig. 2. Measures 4 and 5 of *Dahl polska efter Ola Olsson* as a chronotonic sequence

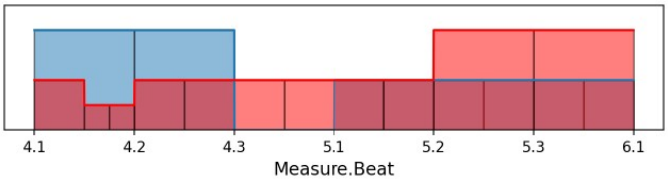

Fig. 3. Overlaid chronotonic sequences for measures 4 and 5 of *Dahl polska efter Ola Olsson* and *1814*.

low enough to forgive the occasional abnormal edit. Under this framework, a "good" Levenshtein distance corresponds to a lower number, which indicates similar tonal profiles and pitch interval sizes (as captured by operation weights) as the Slangpolska scores. A "bad" Levenshtein distance corresponds to a higher number, which indicates less overlap in the qualities listed above.

Our formal method for evaluating pitch differences, then, is the Slangpolska-edit-operation-incidence-weighted Levenshtein edit distance between the first eight measures of scores A and B ($L_{A,B}$). If

$$L_{j,k} = \begin{cases} L_{j,k-1} + i_k & \text{if } A_j = 0 \\ L_{j-1,k} + i_j & \text{if } B_k = 0 \\ L_{j-1,k-1} & \text{if } A_{j-1} = B_{k-1} \\ min[L_{j-1,k} + i_j, & \\ L_{j,k-1} + i_k, & \\ L_{j-1,k-1} + s_{j,k}] & \text{otherwise,} \end{cases} \quad (1)$$

where $L_{j,k}$ is the re-weighted Levenshtein edit distance between the pitch-string of score A ending at position j and the pitch-string of score B ending at position k, i_j is the cost of inserting (or deleting) element j of score A's pitch-string, i_k is the cost of inserting (or deleting) element k of score B's pitch-string, and $s_{j,k}$ is the cost of substituting element j of score A's pitch-string with element k of score B's pitch-string (or vice-versa), then $L_{A,B} = L_{|A|,|B|}$.

The second part of our overall distance representation is **rhythm**. Toussant claims chronotonic distance is a particularly useful transformation-based method for describing the relationship between families of rhythms [9]. We select chronotonic distance as our evaluation metric for its status as a transformation-based method, its endorsement in the literature, and its ability to model what we consider the most important quantifiable aspects of rhythm: onsets, durations, and metrical structure.

To define chronotonic distance, we must first define a chronotonic sequence. A rhythm's chronotonic sequence is a series of squares organized left to right on a 2-dimensional plane. Each square represents a note; a square's left edge corresponds to a note's onset location, while a square's height and width correspond to note duration. Empty space in a chronotonic sequence corresponds to a rest. Other elements (including pitch) are not measured. Figures 1 and 2 display the same two measures as musical score and as a chronotonic sequence respectively. Note that on a longer timeframe, onset patterns reveal the underlying metrical grid in a chronotonic sequence.

We define the chronotonic **distance** between two scores as the sum of all geographic space occupied by one and only one chronotonic sequence when two chronotonic sequences are overlaid on the same graph. For example, Figure 3 overlays the chronotonic sequences of two scores, while Figure 4 illustrates their geographic disparity; the sum of the area shaded grey in Figure 4 is the chronotonic distance between measures four and five of the plotted scores. Under this framework, a "good" chronotonic distance corresponds to a lower number, which indicates similar onsets, durations, rests, and underlying metrical schemata. A "bad" chronotonic distance corresponds to a higher number, which indicates less overlap in the qualities listed above. While a distance of 0 corresponds to perfect rhythmic equality, it does not imply perfect equality in general, as certain unmeasured and irrelevant qualities (notably pitch) may still differ.

We may formalize the chronotonic distance between the first 8 measures of score A and score B ($D_{A,B}$) as:

$$D_{A,B} = \int_1^9 |C_A - C_B| dm \quad (2)$$

where m refers to measure number and C_A and C_B refer to the piecewise horizontal lines which trace the tops of chronotonic chain squares for scores A and score B respectively.

The third element of our overall distance metric involves **form**. Our form distance metric quantifies and aggregates pairwise comparisons of individual measures between two scores in a bipartite graph framework. Our thought process is as follows: the more measures in score A for which we can uniquely assign a similar measure in score B, the stronger the alignment in material repetition, subdivision of parts, metrical behavior, and formal functions between the two scores. We argue that our distance metric captures structural similarities rather than fleeting similarities because (1) each measure is uniquely paired with one and only one measure from the opposite score and (2) all pairings are weighted equally.

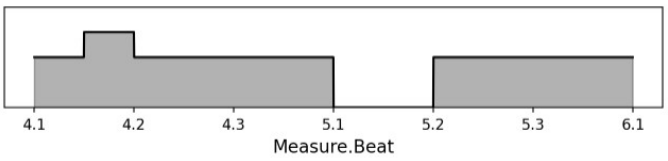

Fig. 4. Geographic difference in chronotonic sequences for measures 4 and 5 of *Dahl polska efter Ola Olsson* and *1814*. **The sum of this area is equal to distance.** In this case, the distance is 2.625 (note that it is the length of a quarter note which receives a geographic value of 1).

First, for each measure from the Slangpolska corpus, we organize all N constituent notes or rests into a string of the following form: "*onset1_pitch1 onset2_pitch2 ... onsetN_pitchN*". We then apply to these measure representations FastText, a model which trains Skipgram embeddings on n-gram subdivisions of each input's tokens, to train 128-dimensional embeddings for use with *onset_pitch* tokens and, by extension, measures composed of these tokens [10]. We select FastText for its ability to embed previously unseen tokens due to consideration of token subdivisions - a behavior which should, for example, enable Random score embeddings to reflect the importance of certain onset locations (as learned from the Slangpolska corpus) even if an onset location is paired with a novel pitch.

We define the form distance between the first eight measures of two scores A and B ($F_{A,B}$) as the "lowest cost" way to pair off each measure in score A with a single measure in score B **without assigning a measure in B to multiple measures in A**, where "cost" refers to angular distance between two measrues' FastText embeddings. Formally, if we let

$$AD_{A_i,B_j} = \frac{\arccos(\frac{V_{A_i} \cdot V_{B_j}}{||V_{A_i}||||V_{B_j}||})}{\pi}, \tag{3}$$

where V_{A_i} and V_{B_j} are the 128-dimensional vector embeddings for measure i of score A and measure j of score B respectively, then

$$F_{A,B} = \min \sum_{i=1}^{8} \sum_{j=1}^{8} AD_{A_i,B_j} P_{A_i,B_j}, \tag{4}$$

where P is a square boolean matrix, P_{A_i,B_j} equals 1 if measure i in score A is paired with measure j in score B and 0 otherwise, and the values in each row of P sum to 1. We use the so-called "Hungarian method" [11] as implemented in the Python library SciPy [12] to arrive at the pairing matrix P which achieves a minimum $F_{A,B}$. A "good" $F_{A,B}$ corresponds to a lower number, which indicates similar musical motifs, metrical behavior, form functions, and subdivision - provided the embeddings accurately target this information. As a sense check, we present a measure-level self-similarity matrix (5) for *Ninas slangpolska* (full score in Appendix Figure 1, note-and-rest-level self-similarity matrix in Appendix Figure 2); brighter colors correspond to lower angular distances between embeddings, while darker colors correspond to higher angular distances embeddings.

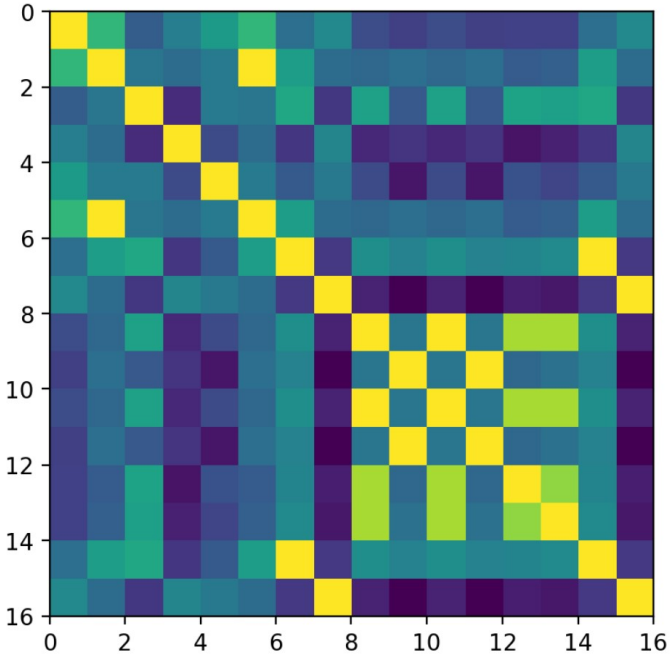

Fig. 5. Self-similarity matrix for the measures in *Ninas slangpolska* (brighter = lower angular distance between embedded measures). See Appendix for full score and event-level self-similarity matrix.

As expected, the diagonal corresponds to a series of angular distances equal to 0, reflecting exact copies of measures and musical events. The still-bright squares representing the interaction between measures 1 and 2 (see Figure **??** for reference) expresses short-term dependency, with little distance created by splitting and shifting the first measure's second note. The bright checkered pattern between measures 9 and 14 captures an alternating motif with pitch transposition, while the dark ring encircling these measures indicates medium-term dependency and collective membership in a subdivision of the score. On the longest timescale, the final measure's relatively low angular distance to the first half of measures as opposed to the second half of measures appears partly due to long-term closure given the score's stable return to its tonic. In sum, while measure embeddings may appear unreadable to the human eye, we believe Figure 5 is indicative of their ability to capture information about musical structure, justifying our $F_{A,B}$ construction.

Note that Equation 4 can be thought of graphically: if we recreate Figure 5 with a different score on each axis and extract the eight-by-eight matrix at the top left of the full matrix, form distance is equal to the minimum possible sum across all sets of angular distances wherein each self-similarity matrix row contributes one angular distance to any given set.

We calculate $L_{A,B}$, $D_{A,B}$, and $F_{A,B}$ for all A and B such that:

1) A is one of 512 polska scores and B is one of 511 other polska scores.
2) A is one of 512 polska scores and B is one of 500 bigram-generated scores.
3) A is one of 512 polska scores and B is one of 500

randomly generated scores.

4) A is one of 500 bigram-generated scores and B is one of 499 other bigram-generated scores.
5) A is one of 500 bigram-generated scores and B is one of 500 randomly generated scores.
6) A is one of 500 randomly generated scores and B is one of 499 other randomly generated scores.

These six unique combinations represent all possible distances due to our symmetric distance functions.

To get a more complete representation of overall distance between two scores, we normalize all $L_{A,B}$, $D_{A,B}$, and $F_{A,B}$ by subtracting from each set their respective minimum and dividing by their respective range. Next, we calculate $M_{A,B}$, our ultimate score-to-score distance metric, through a simple weighted average of our pitch, rhythm, and form distances:

$$M_{A,B} = \frac{1}{3}L_{A,B} + \frac{1}{3}D_{A,B} + \frac{1}{3}F_{A,B} \tag{5}$$

We investigated a variety of ways to represent $M_{A,B}$: L1-norm, L2-norm, further p-norms minimized with parameter tuning on p, a Cobb-Douglas inspired product $L^{\alpha}_{A,B}D^{\beta}_{A,B}F^{\gamma}_{A,B}$ minimized with parameter tuning on α, β, and γ, and a simple linear combination $\alpha L_{A,B} + \beta D_{A,B} + \gamma F_{A,B}$ minimized with parameter turning on α, β, and γ. Where possible, parameter tuning yielded values at or near α = 0, β = 1, and γ = 0 (that is, a disregarded $L_{A,B}$ and $F_{A,B}$), a result we deem unsatisfactory. We ultimately decided on a simple linear combination with equal weights assigned to each distance component - the least biased way to aggregate our distance components while still allowing $L_{A,B}$ and $F_{A,B}$ to substantially impact $M_{A,B}$, in our opinion.

The results of our 1,522,632 $M_{A,B}$ calculations are discussed in Section V; we close Section IV with a few final comments on our distance metrics:

Cosine similarity between measure embeddings shares a moderately positive correlation with Levenshtein edit distances for both pitch and rhythm sequences [5]. In other words, $F_{A,B}$ is correlated with $L_{A,B}$ and likely with $D_{A,B}$, as the latter two's musical material is partly baked into $F_{A,B}$'s embeddings. Even so, $F_{A,B}$ contains information our other two submetrics do not possess, while $L_{A,B}$ and $D_{A,B}$ target their intended metrics more closely. Rather than supplant $L_{A,B}$ and $D_{A,B}$ with $F_{A,B}$, then, we combine the three.

Our aggregate distance metric does little to reconcile differing variance between normalized pitch, rhythm, and form distances. Average rhythm distances between Slangpolska scores cluster near the bottom of the total range, while average pitch and form distances are more often in the neighborhood of distances for other set comparisons. The result is an aggregate distance metric which emphasizes pitch and form differences more than rhythm differences despite receiving equal weights in the weighted sum. This is partially alleviated by the fact that our form distance metric is correlated with pitch and rhythm distance, as discussed above.

Finally, we note that chronotonic distance and form distance are partially blind to the reason for disparity. For example, regarding chronotonic distance, the quarter rest just before measure five in Figure 1 is as geographically 'far away' from two eighth notes as the preceding quarter note; both have the same effect on distance. The embedding-based approach to form distance, meanwhile, renders human interpretation of the details of structural disparity difficult even if our 1-to-1 measure pairing approach targets general structure disparities. Nonetheless, these distance metrics remain useful for their ability to target the musical qualities they are designed to target well enough.

V. Results and Discussion

To aggregate our score distances at the set-to-set level, we find the average minimum score distance from each score A in a score set $\mathbb{S}_1$ to scores in score set $\mathbb{S}_2$. We may formulate this as:

$$m_{\mathbb{S}_1,\mathbb{S}_2} = \mathbb{E}_{A\in\mathbb{S}_1}[\min_{B\in\mathbb{S}_2} M_{A,B}] \tag{6}$$

where $(\mathbb{S}_1, \mathbb{S}_2) \in \{$(Actual, Actual), (Actual, Generated), (Actual, Random), (Generated, Generated), (Generated, Random), (Random, Random)$\}$, Actual refers to the Slangpolska scores, Generated refers to the bigram-generated scores, and Random refers to the randomly generated scores.

$m_{\mathbb{S}_1,\mathbb{S}_2}$ values are plotted as red dots in Figure V. Interquartile ranges for the individual $M_{A,B}$ minimums (the values in Equation 6 before taking the expectation) are also provided.

As might be expected, Actual-to-Random and Generated-to-Random comparisons yield the highest minimum score distances, while the Actual-to-Actual comparisons are comparatively low (recall that score distances of 0 are not permitted to enter the data). Lowest by far are distances between scores in the Generated set, suggesting our form-motivated means of control is rather strict.

Comparing Slangpolska scores to our bigram-generated scores yields an average minimum score distance approximately halfway between the Actual-to-Actual and Actual-to-Random averages. In other words, our generative model can emulate about half of the pitch, rhythm, and form behavior that differentiates Slangpolska scores from our randomly-generated scores. Worth noting is that a Student t-test comparison of $m_{Actual,Actual}$ and $m_{Actual,Random}$ differentiates these two means at the 95% significance level, whereas a t-test comparison of $m_{Actual,Actual}$ and $m_{Actual,Generated}$ does not.

There is also cause to investigate pre-expectation minimums at the score level. For example, given a score $A \in Actual$, the probability that $\min_{B\in Generated} M_{A,B} < \min_{B\in Random} M_{A,B}$ is 99%. In other words, provided we have enough scores to train on and we generate enough scores, our generative model is effectively universally capable of outperforming random score generation regardless of the specific 'Actual' comparison score targeted.

As seen in Figure 7, our model adheres to the Slangpolska corpus' overall pitch distribution in some ways but diverges in others. Pitch classes D, E, G show very similar incidence, classes C and A show somewhat similar incidence, and classes

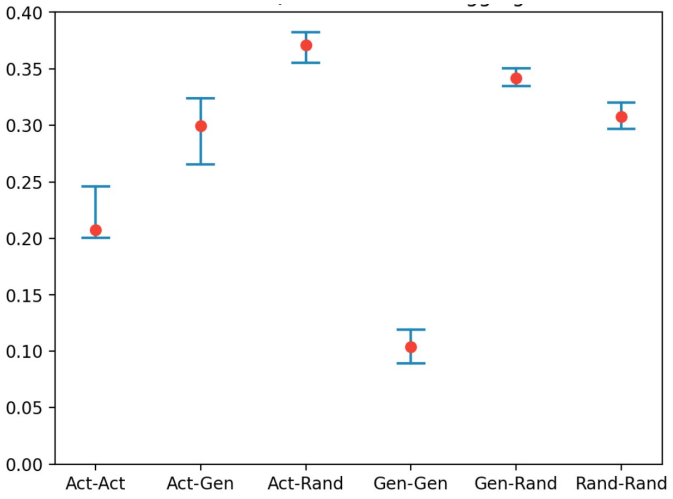

Fig. 6. Set-to-Set means and IQRs for minimum score distance. Act refers to the Slangpolska corpus, Rand refers to the baseline model, and Gen refers to the Bigram model.

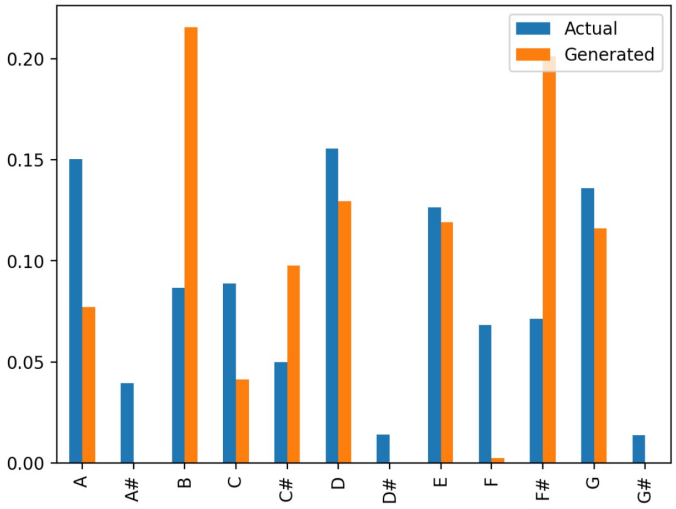

Fig. 7. Global pitch distribution for Slangpolska scores

B and F exhibit large discrepancies. By splitting our Generative model's underlying probability distributions according to key and enforcing patterns of specific keys, it appears we have distorted our pitch distribution away from the Slangpolska corpus in pursuit of proper form.

Upon listening to our generated scores, we find that Narmour's principles [13], such as the proximity principle, aren't frequently respected in either "good" or "bad" scores. For example, in one of our generated sequences which yielded relatively low distances in relation to Slangpolska scores (see Figure 8), several intervals exceed 5 semitones (the second interval of the first measure is one example). This is something we also find in "bad" generations (see Figure 11 for a score that yielded relatively high score distances). That said, our scores aren't entirely devoid of Narmour's concepts. Somewhat comforting is that the largest interval in the first measure of the aforementioned 'good' score adheres to Narmour's registral direction.

Overall, we see that the structural control we introduce into our generative model yields the same picture of long-term dependency in each generated score, as evidenced by the exact same pattern of yellow squares in measure-level self similarity matrices (see 9 and 12 for example; brighter colors indicate lower angular distances between measure embeddings). Though variation is present outside these repeating measures (note the differing pattern of note-embedding comparisons between 10 and 13), we do find our our generated results to be somewhat unvaried. Still, with form functions and repetition pleasant ever-present, form control means even the "bad" generated scores are not unpleasant to listen to. It would seem our model's difficulties with form arise in the dimension of structural variation and controlled melodic contour, as evidenced by the similarity of self-similarity matrices and many scores' inability to finish with a stable return to the score's tonic (see 14 and 15 for a score with a final note dissimilar to earlier notes). With such strong form control, what separates the "good" from the "bad" according to our distance functions tends to be related to rhythm and pitch.

Regarding the generated measures individually, we note that by using bigrams, our generative model selects notes from a probability distribution based only on the previous note. Since probability prefers smaller steps and 'unusual' behavior is quickly correctly due to the low probability of consecutive rare steps, our melodies can appear generally uninspired and unable to commit to adding "spice". Additionally, the control in our model sacrifices the imitative power of our pitch classes, as evidenced by the previously discussed Figure 7.

Our generative model successes to take into account musical structures and motifs, but in a very repetitive way. It also deprives our scores of a stronger relationship between pitch and rhythm reflected in tonal hierarchy. In the future, we could could take into account that strong beats have a tendency to fall on stable notes, or that a leading note is unlikely to fall on a weak beat, by updating our probability distribution after a strong beat and stable note is initially established in a score.

Finally, we think it is important to broaden the pitch possibilities in the model to generate a richer sequence of melodies with contour, direction and line taken into account.

VI. Conclusion

Using bigrams to model musical scores and a mix of chronotonic, Levenstein, and measure embedding distances to quantify our results, we achieve moderate success with regards to distance values and human hearing experiments. We have learned that bigrams are a weakly sufficient tool for emulating some aspects of human-like scores, but that a model informed entirely by the previous note and measure variant permutations cannot generate a truly satisfying musical piece. Future model development should incorporate more memory and be less strict with structural control to generate more natural, varied scores.

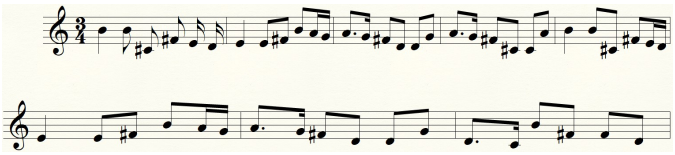

Fig. 8. Good score example (bigram #276)

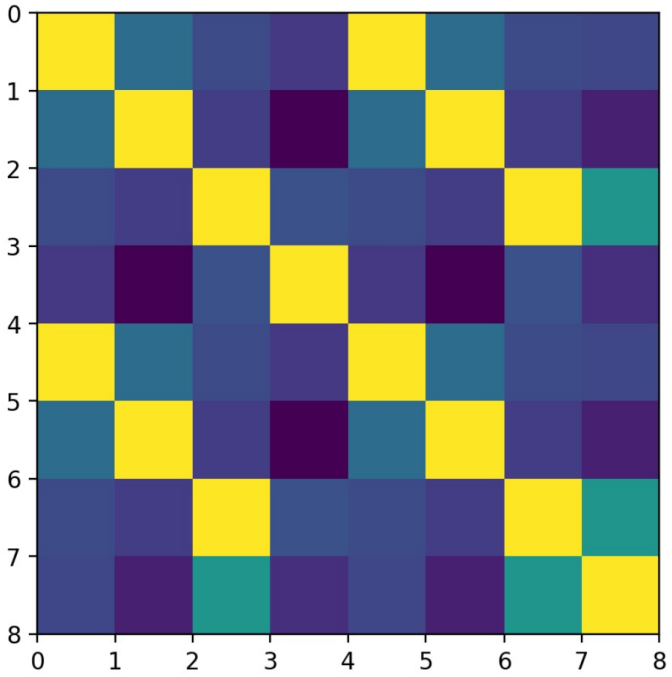

Fig. 9. Good score measure-level SSM (bigram #276)

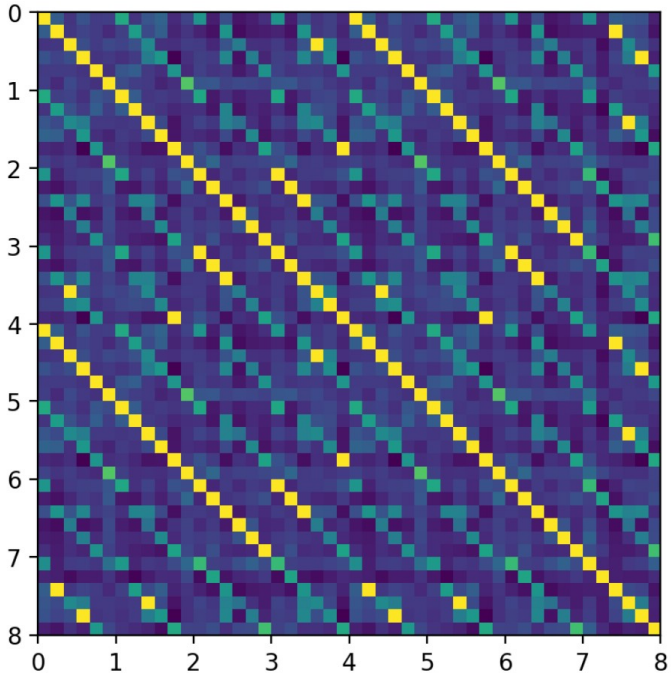

Fig. 10. Good score note-level SSM (bigram #276)

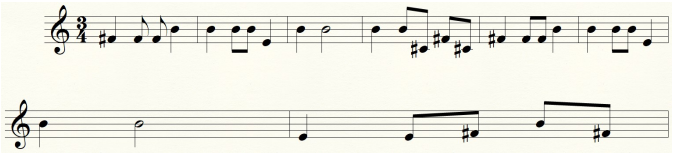

Fig. 11. Bad score example (bigram #452)

REFERENCES

[1] Michael Scott Cuthbert and Christopher Ariza. Music21: A toolkit for computer-aided musicology and symbolic music data. In J. Stephen Downie and Remco C. Veltkamp, editors, *ISMIR*, pages 637–642. International Society for Music Information Retrieval, 2010.

[2] Godfried T Toussaint, Luke Matthews, Malcolm Campbell, and Naor Brown. Measuring musical rhythm similarity: Transformation versus feature-based methods. *Journal of Interdisciplinary Music Studies*, 6(1), 2012.

[3] Juan Beltran, Xiaohua Liu, Nishant Mohanchandra, and Godfried Toussaint. Measuring musical rhythm similarity: Statistical features versus transformation methods. *International Journal of Pattern Recognition and Artificial Intelligence*, 29:1550009, 03 2015.

[4] Matthew Brian Kelly. *Evaluation of melody similarity measures*. Queen's University (Canada), 2012.

[5] Harsh Jhamtani and Taylor Berg-Kirkpatrick. Modeling self-repetition in music generation using generative adversarial networks. In *Machine Learning for Music Discovery Workshop, ICML*, 2019.

[6] Alexandra L Uitdenbogerd. Variations on local alignment for specific query types. *MIREX 2006*, page 24, 2006.

[7] Amaury Habrard, José Manuel Inesta, David Rizo, and Marc Sebban. Melody recognition with learned edit distances. In *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshop, SSPR & SPR 2008, Orlando, USA, December 4-6, 2008. Proceedings*, pages 86–96. Springer, 2008.

[8] Eric Sven Ristad and Peter N Yianilos. Learning string-edit distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5):522–532, 1998.

[9] Godfried Toussaint. A comparison of rhythmic similarity measures. 01 2004.

[10] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.

[11] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[12] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[13] Eugene Narmour. *The analysis and cognition of melodic complexity: The implication-realization model*. University of Chicago Press, 1992.

APPENDIX

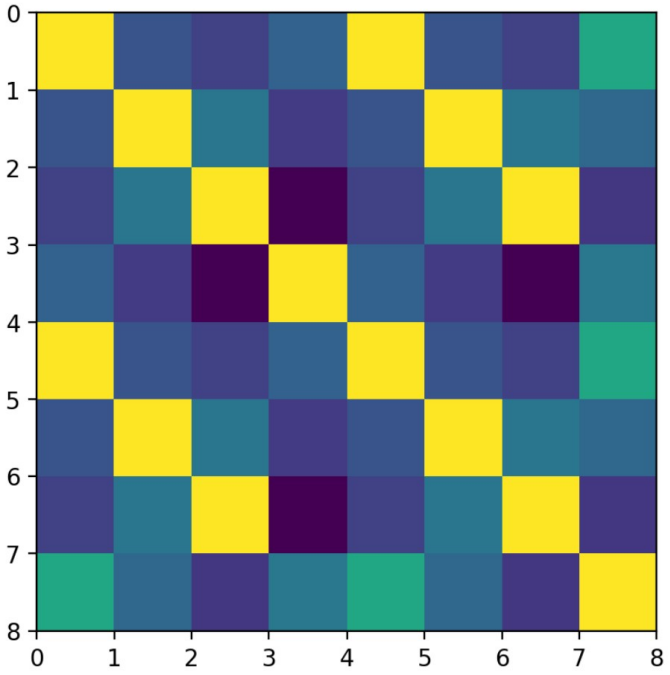

Fig. 12. Bad score measure-level SSM (bigram #452)

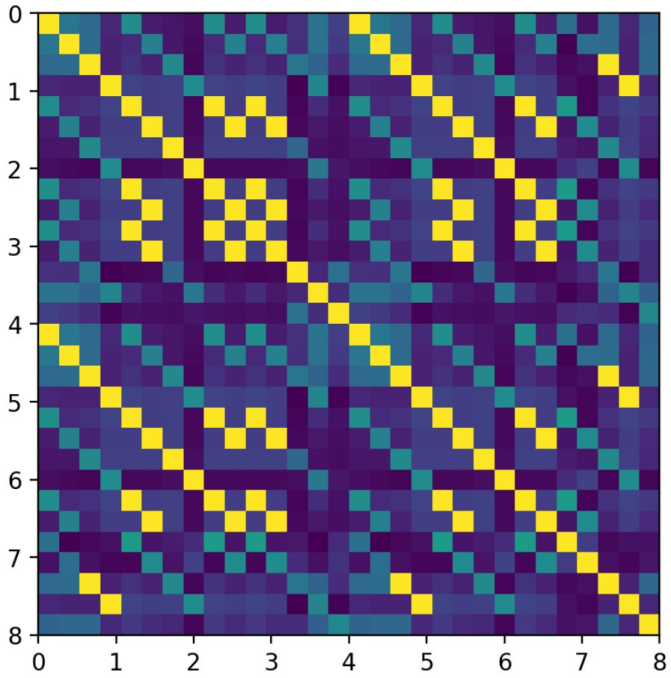

Fig. 13. Bad score note-level SSM (bigram #452)

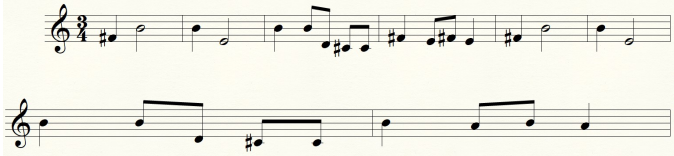

Fig. 14. Bad score example (bigram #362)

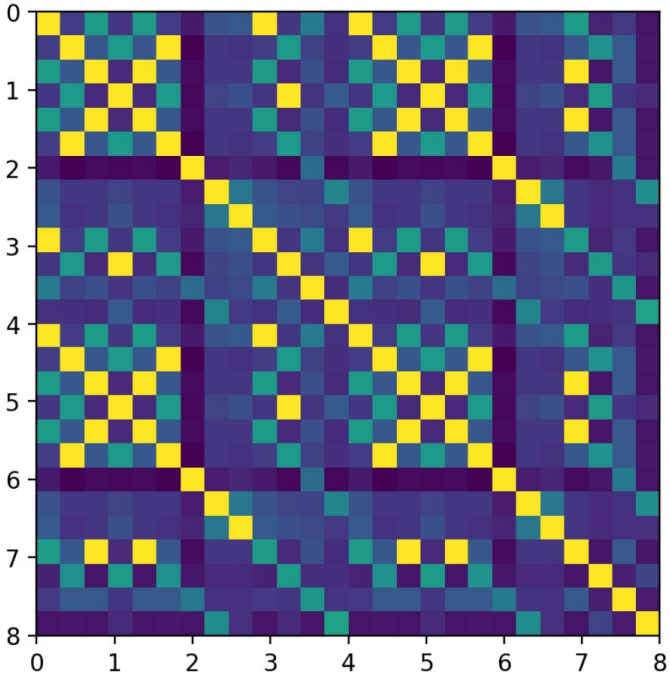

Fig. 15. Bad score note-level SSM (bigram #362)

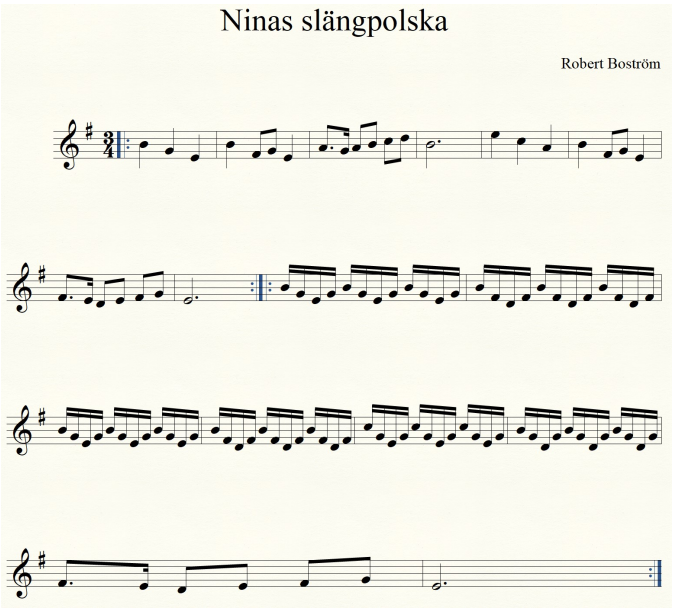

Fig. 1. Full score for Ninas slangpolska

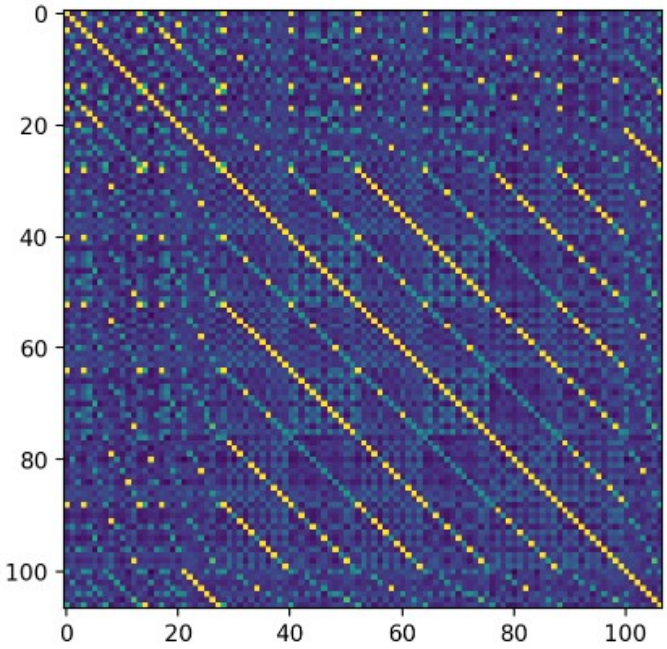

Fig. 2. Self-similarity matrix for the notes and rests in *Ninas slangpolska* (brighter = lower angular distance between embedded events).