SUPPLEMENTARY MATERIALS OF "MELOFORM: GENERATING MELODY WITH MUSICAL FORM BASED ON EXPERT SYSTEMS AND NEURAL NETWORKS"

Peiling Lu[1] **Xu Tan** [1] **Botao Yu**[2]
Tao Qin[1] **Sheng Zhao**[3] **Tie-Yan Liu**[1]
[1] Microsoft Research Asia, Beijing, China
[2] Nanjing University, Nanjing, China
[3] Microsoft Azure Speech, Beijing, China
{peil,xuta,taoqin,szhao,tyliu}@microsoft.com, btyu@smail.nju.edu.cn

1. CHORD PROGRESSION GENERATION

The chord progression generation is to generate chord progression for motif and phrase for helping build note sequences in melodies. The chord progression is built upon harmonic functions [1], which can be classified as tonic (T), subdominant (S) and dominant (D) in common-practice music. Figure 1 illustrates the harmonic progression flowchart in major scale. The high-level harmonic progression flows as T-S-D-T. In each functional group, the chords can progress to each other as shown in dotted line. The progression for the motif and phrase can be constructed with this pattern. Particularly, at the end of each phrase and sections, cadence [2] is considered when building chord progressions. There are two types of cadences we used: authentic cadence (i.e. chords ending with dominant to tonic chord) and half cadence (i.e. chords ending with tonic chord). Phrases at the end of sections can end with authentic cadence, while phrases inside sections can end with both of them. This depends on how you develop your music ideas. Furthermore, to narrow down the selection range, we collected some best-seller progressions to construct a n-gram distribution for helping direct the chord progression.

2. RHYTHM PATTERN GENERATION

The rhythm pattern generation creates the rhythm patterns for the motif. For example, We randomly generate the rhythm patterns for one-measure motif in 4/4 meter as follows: 1) We divide the 4-beat measure into uniform eight positions (i.e. one quarter note is partitioned into two eighth notes) as one chunk. 2) We randomly select a subset of this eight positions to insert n rest notes, which divides the original chunk into $n+1$ chunks. 3) In each chunk, we randomly select the subset of the positions to insert boundaries to determine which set of notes need to be merged. The notes between two consecutive boundaries are bound to be joined into a longer note. There are some limitations to make the rhythm applicable for singing: 1) The accumulated duration of rest notes in one measure is not greater than two beats. 2) The duration of each rest note is not greater than one beat. 3) The duration of each note is not greater than four beats. 4) The number of notes in each measure is not less than three.

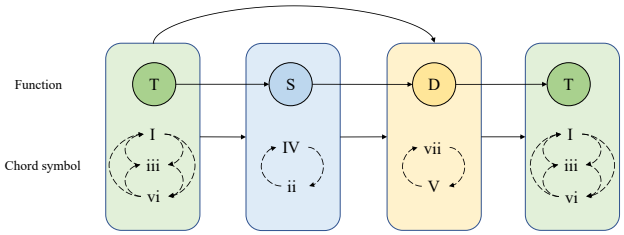


Figure 1. Transformer based Melody Polisher

3. DEVELOPMENT STRATEGY

In order to develop the motif into the phrase, we define three categories of development strategies: sequence, transformation and ending. We will illustrate the specific methods used in each strategy.

Sequence we leverage the sequence [3] to restate the motif in development bars, such as real sequence, tonal sequence, rhythmic sequence and modified sequence. The specific one is chosen by referring to the chord in development bars. For example, if the chord progression of development bars is the same as motif, we can randomly choose the methods above without any limitations. However, if the chord progression changes, pitch selection should also be changed for harmony, rhythm sequence is a better choice for this situation.

[1] https://musictheory.pugetsound.edu/mt21c/HarmonicFunction.html
[2] https://en.wikipedia.org/wiki/Cadence
[3] https://en.wikipedia.org/wiki/Sequence_(music)#Melodic_sequences

Transformation For transformation, we propose some methods of acceleration, decoration, fragmentation, and fine-tuning. As shown in Figrue 2(a), Acceleration speeds up the motif by reducing note duration and inter-onset interval (IOI)[4]. This may results in bar count mismatch since notes are move ahead, sequence can help fill the remaining measures. Decoration in Figure 2(b) is the same with that in motif generation. Fragmentation in Figrue 2(c) is to randomly select a fragment of the motif, and sequence it to the desired length for emphasizing the motif once again. Fine-tuning is to change some notes for motif variation by moving the pitch, merging notes, splitting notes, decorating notes, etc.

Ending Ending is to give a sense of the end, which can be achieved by generating a new melody using the same method of motif generation, and force the last pitch lower than the first pitch to form a downward melody shape. Another way is to prolong the note length in the last bar for a feeling of rest.

4. PHRASE BOUNDARY AND SIMILARITY DETECTION ALGORITHM

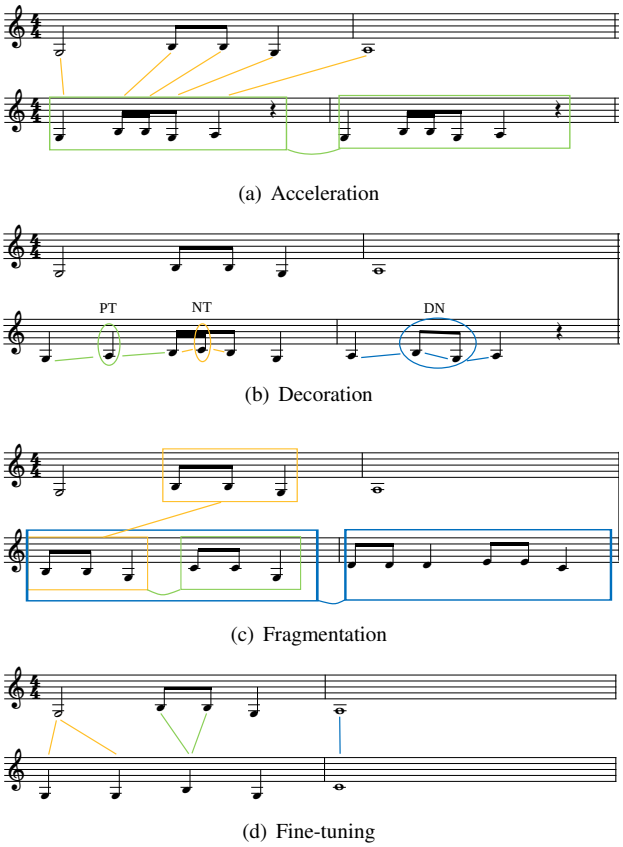
The phrase boundary has been detected by calculating the onset intervals between adjacent notes. If this onset interval is greater than 1.5 beats, and there exists a rest between notes, we recognize there exists a boundary between these two segments. The phrase similarity is calculated by string edit distance. Saying that there are token sequences of two phrases, we use the string edit distance to calculate the similarity degrees, then regard two phrases are similar if the degree is greater than 0.9.

5. CASE STUDY

Figure 3 shows an example of a generated melody from MeloForm. The musical form for this melody is represented as $A(a_1, a_1^{'}, a_1^{''})B(b_1, b_1^{'}, b_1^{''})A(a_1, a_1, a_1)$. In phrase a_1, 1-bar motif is developed into 4-bar phrase by sequence, sequence and ending. Combining with phrases $a_1^{'}$ and $a_1^{''}$, the variations from phrase a_1, we form the section A. For phrase b_1, motif is developed by transformation, sequence, and a compound of ending and transformation. Phrase b_1 and its variations $b_1^{'}$ and $b_1^{''}$ form section B. Section A recurs again with three phrases a_1. More examples are uploaded to demo page[5].

6. POP909_LM

We trained the language model using Transformer as the backbone. The input sequence is encoded the same with MeloForm except that the phrase label is added for each note. When predicting melodies given musical form, we enforce the model to place the given label at the beginning of the phrase, which conditions the model to predict the note sequence for the given phrase.



(a) Acceleration

(b) Decoration

(c) Fragmentation

(d) Fine-tuning

Figure 2. Transformation strategies. Acceleration: lines in yellow describe the length shortening for each note, while yellow lines illustrate sequence to the next bar. Decoration: yellow notes represent Passing Tone, blue notes represent Complete Neighbor Tone, and blue notes represent Double Neighbor Figure. Fragmentation: motif in yellow box is extracted to be the fragment to be developed. This fragment is firstly sequenced to fill up the remaining bar, and the note sequence from this bar will be sequenced to the next bar for matching the number of bars in original motif. Fine-tuning: yellow lines represent the way to split the notes; green lines represent the method for merging notes; blue lines represent moving pitch.

[4] https://en.wikipedia.org/wiki/Time_point
[5] https://meloform.github.io/

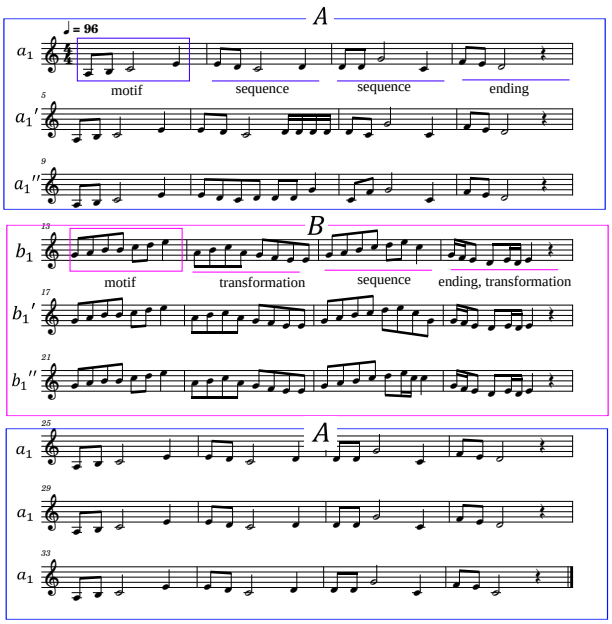


Figure 3. Case study example.