Cluster Based Melody Generation

Karishma Mahajan , Shivani Jangam, Juhi Gandhi , Pradnya Kale

BE(Computer Engineering),Sinhgad Academy of Engineering,Pune

Prof: Kiran Avhad, Dept. of Computer Engineering, Sinhgad college, Maharashtra, India

---***---

Abstract - *This project describes the development of an application for generating tonal melodies. The goal of the project is to ascertain our current understanding of tonal music by means of algorithmic music generation. The method followed consists of four stages: 1) selection of music-theoretical insights, 2) translation of these insights into a set of principles, 3) conversion of the principles into a computational model having the form of an algorithm for music generation, 4) testing the "music" generated by the algorithm to evaluate the adequacy of the model. As an example, the method is implemented in Melody Generator, an algorithm for generating tonal melodies. The program has a structure suited for generating, displaying, playing and storing melodies, functions which are all accessible via a dedicated interface. The actual generation of melodies, is based in part on constraints imposed by the tonal context, i.e.* ***by*** *meter and key, the settings of which are controlled by means of parameters on the interface. Out proposed system will add parallel processing activities to get output in very short time.*

Key Words: Melody, Music Generation, Music-theoretical Insights, Melody Generator, Tonal Melodies.

1.INTRODUCTION

Most of the approaches above exhibit very simple representations in an attempt to decrease the search space, which in some cases compromises their output quality. Musical questions are sometimes left unanswered, too. how can we expect to evaluate the system's response if we do not have a harmonic context for it? Melody is a series of musical notes arranged in succession, in a particular rhythmic pattern, to form a recognizable unit. Any melody is a composition of permutation and combination of only 7 notes, sa re ga ma pa dha ni. To create an application where we will help music composers to create a unique melody. The major problem is computation power required to do such operation. This project describes the development of an application for generating tonal melodies. The goal of the project is to ascertain our current understanding of tonal music by means of algorithmic music generation. The method followed consists of four stages: 1) selection of music-theoretical insights, 2) translation of these insights into a set of principles, 3) conversion of the principles into a computational model having the form of an algorithm for music generation, 4) testing the "music" generated by the algorithm to evaluate the adequacy of the model. As an example, the method is implemented in Melody Generator, an algorithm for generating tonal melodies. The program has a structure suited for generating, displaying, playing and storing melodies, functions which are all accessible via a dedicated interface. The actual generation of melodies, is based in part on constraints imposed by the tonal context, i.e. by meter and key, the settings of which are controlled by means of parameters on the interface. Out proposed system will add parallel processing activities to get output in very short time. The basic objective of project is of what constitutes a melody, and in particular a beautiful melody, is constantly changing in music. Music composers have to invest much amount of time and efforts to create a beautiful melody. We wanted to develop a system which will apply rhythmic fitness rules to filter good melody from the randomly generated melody sequence. This will help music composer to create a good melody in very short time as per his/her need.

1.1 LITERATURE SURVEY

Music: A Melody Composer based on Frequent Pattern Mining

There are a bulk of studies on proposing algorithms for composing the melody of a song automatically with algorithms, which is known as algorithmic composition. However, within a song, there usually exists a certain extent of correlation between its melody and its lyric. Hence, we need a new melodycomposition algorithm and a melody composer which employs this composition algorithm. When a lyric is present in a song, algorithmic composition should consider not only the temporal correlation among all notes (or sounds) of the melody in the song but also the lyricnote correlation between the notes and the lyrics in the song.

Genetic algorithms in algorithmic composition have a short but interesting history, largely lying in the school of style replication. John A. Biles utilized genetic algorithms to generate jazz solos. In, a genetic algorithm for making music compositions is presented. Position based representation of rhythm and relative representation of pitches, based on measuring relation from starting pitch, allow for a flexible and robust way for encoding music compositions. This approach includes a pre-defined rhythm applied to initial population, giving good starting solutions. Modified genetic operators enable significantly changing scheduling of pitches and breaks, which can restore good genetic material and prevent from premature convergence in bad suboptimal solutions. Beside main principles of the algorithm and methodology of development, in this paper, some solutions are presented in the musical score.

1.2 ALGORITHM

Minimum the time span probability stays 50-50, but as time span increases the probability of that thing being abandon or hazardous increases. Hence the incident can be avoided in that case. Input : Melody notes & Duration in sec Output: All Possible combination of Melodies in the form of array Algorithm Steps: For each melody in the training set Do

1. Validate if the duration is valid as per input notes
2. Extract the musical notes sequence from the input format
3. Split the sequence into individual musical notes
4. Map musical notes with unique character e.g : Sa ==> 'C'
5. Create all possible permutation using 'Recursive Tree' method
6. Find out how many clients are present in cluster
7. Divide the total input sargams into clients
8. Connect to each client and pass its sargams
9. Client will apply rules on each sargam
10. Client will send back all valid sargams
11. Server will wait until it get response from all clients

12. Show the top 100 sargams into result dropdown

End.

1.3. RULES

1) Scale Rule :This rule calculates if a given melody adheres to a specific scale, for example C major (C D E F G A B). The scale can be chosen by the user.

2) Adjacent Notes Rule :It calculates and verifies whether the difference between most adjacent notes (around 70%) is not more than a step and whether leaps are limited to less than 4 semitones.

3) Proportion between Rests and Notes: It finds the proportion between the notes and rests in a given melody. The proportion of notes/rests is set as a parameter.

4) Repeating Notes Rule :If a given melody has repeating notes or rests, it must be observed that it does not exceed a threshold, lest the melody may sound repetitive and boring. The maximum number of repeating notes and/or rests can be set as parameters.

5) Global Pitch Distribution Rule :This rule validates if the lowest and highest pitch of a given melody fall within the margins specified. The margin is indicated as the number of semitones.

1.4 FLOWCHART

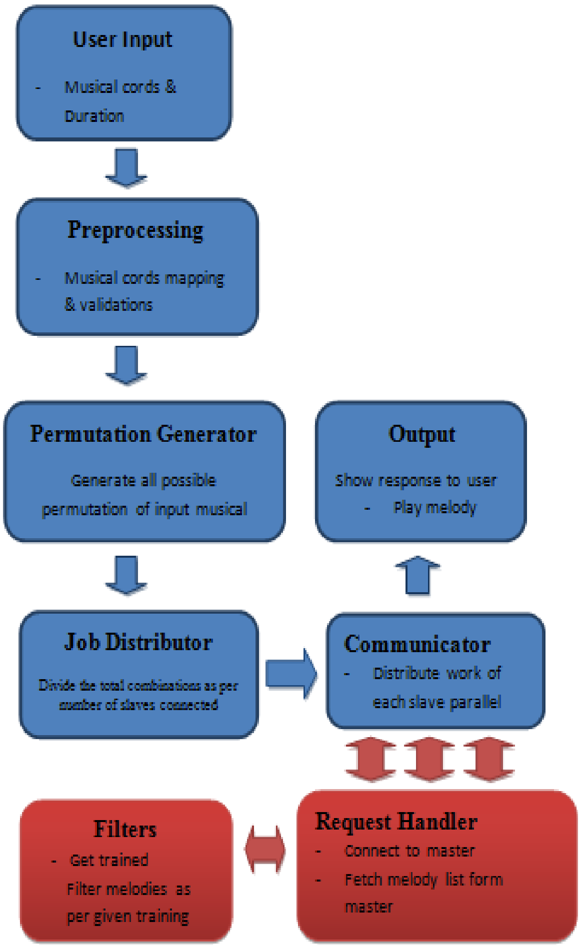

Fig -1: Flowchart

2. CONCLUSIONS

In a system for automatic generation of harmony and accompanying melody in a functional setting, designed to be simple and easy to understand and improve. First, in our melody composition algorithm, to utilize the lyric-note correlation information, which is captured by frequent patterns, we choose to build a Probabilistic Automaton. However, we believe that Probabilistic Automaton is not the only option. In fact, one can consider exploring other models for this purpose. We've seen how our system can generate simple but pleasing pieces, and how it can be modified to support different harmonies and melody styles.

ACKNOWLEDGEMENT

We take this opportunity to thank our project guide Prof. Kiran Avhad for their guidance and providing all necessary facilities for this paper.

REFERENCES

[1] Cheng Long, Raymond Chi-Wing Wong, Raymond Ka Wai Sze, The Hong Kong University of Science and Technology "T-Music: A Melody Composer based on Frequent Pattern" FSGRF12EG50

[2] Yazawa, Sakurako, Masatoshi Hamanaka, and Takehito Utsuro (2014.). Melody generation system based on a theory of melody sequence, Advanced Informatics: Concept, Theory and Application (ICAICTA), 2014 International Conference of. IEEE

[3] Salamon, Justin, et al. (2014) Melody extraction from polyphonic music signals: Approaches, applications, and challenges." Signal Processing Magazine, IEEE31.2: 118-134.

[4] Jos´e Pedro Magalhaes, Department of Computer Science, University of Oxford, Hendrik Vincent Koops, Department of Information and Computing Sciences, Utrecht University , Functional Generation of Harmony and Melody , EPSRC EP/J010