



Figure 2: Gathering Evaluation

to 5. Other structures are negative examples. This gives a generalized structure whose evaluation is better than 5 by each adjective pair. This condition earns 5 points for the adjective pair.

- Similarly, find a condition to accomplish evaluation better than 4. This condition earns 4 points.

The condition for the opposite adjective, such as *dark*, *unfavorable* and *unstable*, earns $6 - g$ points, where g is the grade given by the user. Since 75 pieces are too many to be evaluated in one session, the subjects evaluate them in multiple sessions by comparing a pair of some chosen pieces multiple times.

Each rule is described by a predicate rather than an attribute, since it is hard to describe a score by using only some attributes. PROLOG describes each condition, whose predicates are defined in background knowledge (Numao, Kobayashi, & Sakaniwa 1997). We prepare the following predicates in PROLOG to describe a musical structure, where `frame` is the name of predicate and `/1` is the number of arguments:

- `frame/1` represents the whole framework of music, i.e., tonality, rhythm and instruments.
- `pair/2` represents a pattern of two successive chords.
- `triplet/3` represents a pattern of three successive chords.

For example, we can describe that a subject likes a piece whose tonality is E major or E minor, tempo is Allegretto, accompanying instrument is piano, rhythm is 4/4, and contains a specified pair of successive chords.

To acquire such conditions, we use Inductive Logic Programming (ILP), which is a machine learning method to find

- SubjectA dark
 - `frame(S) :- tonality_moll(S), tempo_larghetto(S).`
 - `triplet(C1, C2, C3) :- moll(C1), form_V(C2), chord_VI(C2), chord_V(C1), inversion_Zero(C3), form_VII(C3).`

Figure 3: Acquiring predicates.

a PROLOG program. A score is represented by a symbol, where a relation between two notes are important. These mean that ILP is a good tool for generalizing a score. Figure 3 shows a score and its generalization described in PROLOG. The variables `C1`, `C2` and `C3` represent successive bars. These clauses mean that *SubjectA* feels a piece dark when its tonality is moll (minor), its tempo is larghetto, the first chord is moll V, the second is triad (`form_V`) VI, and the third is 7th root position (`inversion_Zero`) chord.

Arrangement

The authors constructed the arranger and the composer separately, since arrangement is easier than composition, i.e., the composer is much slower than the arranger. The following method arranges a piece by minimally changing its chord sequence to cause the required feeling:

- Analyze the original chords to recognize their function, e.g., tonic, dominant, subdominant, etc.
- Modify each chord to satisfy the acquired conditions without changing its function.
- Modify the original melody minimally to fit the modified sequence of chords.

This is accomplished by the following windowing procedure:

- Set a window on the first three chords.
- Enumerate the all chords with the same function to satisfy the acquired predicates `pair` and `triplet`. Sum up the points of acquired predicates to evaluate each chord sequence.
- Shift the window by two, i.e., set a new window on the last chord and its two successors. Enumerate the chords similarly to the above.
- Repeat the above to find a sequence with the most points.
- Repeat the above for the all 12 tonality. Determine the tonality that earns the most points.
- Determine the frame that earns the most points.

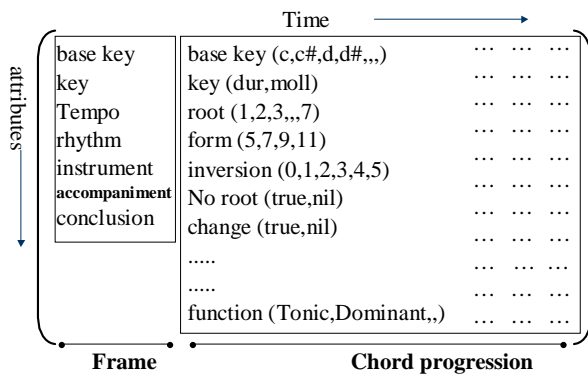


Figure 7: Genotype

This makes possible to produce a chord progression that fits the theory and causes the required feeling. $Fitness_Builtin(M)$ is a fitness function based on the theory of tonal music, which issues a penalty to a chord progression violating the theory. $Fitness_User(M)$ is based on the extracted musical structures that reflect the subject's feelings:

$$\begin{aligned}
 Fitness_User(M) = & Fitness_Frame(M) \\
 & + Fitness_Pair(M) \\
 & + Fitness_Triplet(M)
 \end{aligned}$$

where $Fitness_Frame(M)$ is fitness based on tonality, rhythm and instruments, etc. $Fitness_Pair(M)$ and $Fitness_Triplet(M)$ are based on two or three successive chords, respectively.

For producing a piece, the system uses MACS (Tsunoda 1996), which generates a melody from a chord progression and some rules for the duration. Since MACS is a black box containing complicated program codes, the authors start a new project to find simple rules describing the process, which clarifies the process of generating a melody.

Figure 8 and 9 show created pieces. Figure 8 is a piece the system tried to make bright. Figure 9 is one it tried to make dark. These examples show that the system composes a bright piece without handcrafted background knowledge on brightness and by automatically acquiring some musical structures that cause a bright feeling. Other created pieces are shown in (Numao, Takagi, & Nakamura 2002).

Figure 10 shows evaluation of the composed pieces. '+' shows the average result of pieces the system tried to make positive. '-' shows that it tried to make negative. According to Student's t-test, they are different for 4 adjective pairs at the level of significance $\alpha = 0.05$. They are different for 2 pairs at the level $\alpha = 0.01$. Figure 11 shows the effect of melody, which is dramatic in some adjective pairs.

This system is profoundly different from other composing systems in that it composes based on a personal model extracted from a subject by using a machine learning method. A composing system using an interactive genetic algorithm (IGA), such as GenJam (Biles 2002), may be similar method to ours in that it creates a piece based on the user interaction.



Figure 8: A created bright piece

However, IGA generally requires far more interactions than ours, which reduces the number of interactions by utilizing a personal model generalized from examples, although the detailed comparison between GenJam and ours is a future work. Other advantages are that we can recycle a personal model in many compositions, and manually tailor a predicate in the system to improve its performance.

Related Work

In algorithmic music composition, a simple technique involves selecting notes sequentially according to a transition table that specifies the probability of the next note as a function of the previous context. Mozer (1994) proposed an extension of this transition table approach using a recurrent autopredictive connectionist network. Our system is more flexible than this in that the user specifies an adjective to change impressions of a created piece.

Wiggins (1999) proposed to apply genetic algorithms to music composition. Our method combines a genetic algorithm with a personal model acquired by machine learning.

Widmer (1994) proposed a method of accomplishing explanation-based learning by attaching harmonies — chord symbols to the notes of a melody. The present paper further discusses a means of controlling the process based on learned feelings.

Hirata (1999, 1996) constructed a reharmonizing and arranging system based on a knowledge representation in Deductive Object-Oriented Databases (DOOD). Our system is different in adaptation mechanism by acquiring a personal model.

Thom (2000) proposed to apply unsupervised learning to interactive Jazz/Blues improvisation. In contrast, our method is an application of inductive learning, i.e., supervised learning. Hörnell's system produces and harmonizes



Figure 9: A created dark piece

simple folk style melodies based on learned musical structure (Hörnel & Ragg 1996). Dannenberg, Thom and Watson (1997) apply machine learning techniques to musical style recognition. Our method is different from them in its emotional-driven generation of music.

The Wolfgang system utilizes emotions to enable learning to compose music (Riecken 1998). It is an interesting research topic to compare its cultural grammar and our PROLOG rules based on the semantic differential method. Emotional coloring (Bresin 2000) is an interesting research in the field of automatic music performance with a special focus on piano, although automatic composition is out of its scope.

Conclusion

Pat Langley (1998) proposed an adaptive user interface to be applied to a navigation system (Rogers, Fiechter, & Langley 1999). Our method extends the concept of adaptive user interfaces in a sense that it constructs a new description adaptively. That is why we call our system a *constructive* adaptive user interface.

Acknowledgements

The authors would like to thank Pat Langley and Dan Shapiro, who gave fruitful comments, when one of the authors gave a talk at Center for the Study of Language and Information, Stanford University.

References

- Biles, J. A. 2002. Genjam. <http://www.it.rit.edu/~jab/GenJam.html>.
 Bresin, R. 2000. *Virtual Virtuosity*. Ph.D. Dissertation, Kungl Tekniska Högskolan, Stockholm.

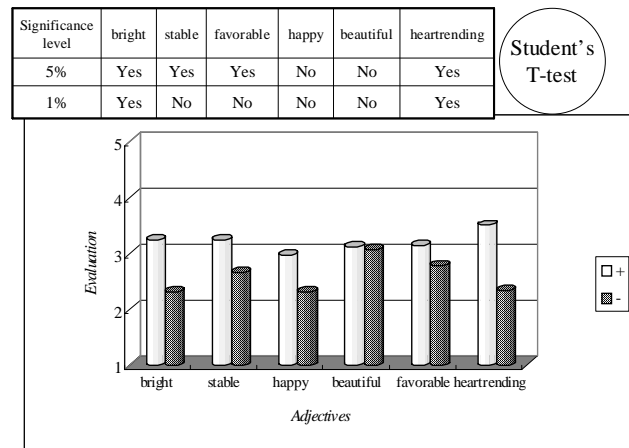


Figure 10: Evaluation of Composition

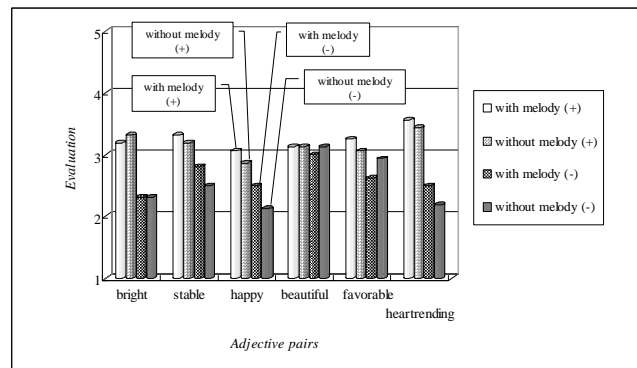


Figure 11: Effects of melodies

Dannenberg, R. B.; Thom, B. T.; and Watson, D. 1997. A machine learning approach to musical style recognition. In *Proc. ICMC97*.

Hirata, K., and Aoyagi, T. 1999. Musically intelligent agent for composition and interactive performance. In *Proc. ICMC*, 167–170.

Hirata, K. 1996. Representation of jazz piano knowledge using a deductive object-oriented approach. In *Proc. ICMC*.

Hörnel, D., and Ragg, T. 1996. Learning musical structure and style by recognition, prediction and evolution. In *Proc. ICMC*. International Computer Music Association.

Langley, P. 1998. Machine learning for adaptive user interfaces. In *CSLI-Stanford University IAP Spring Tutorials*, 155–164.

Michalski, R. S., and Tecuci, G., eds. 1994. *Machine Learning: A Multistrategy Approach (Vol. IV)*. San Francisco, CA: Morgan Kaufmann.

Mozer, M. 1994. Neural network music composition by prediction: Exploring the benefits of psychoacoustic constraints and multiscale processing. *Connection Science*.

- Numao, M.; Kobayashi, M.; and Sakaniwa, K. 1997. Acquisition of human feelings in music arrangement. In *Proc. IJCAI 97*, 268–273. Morgan Kaufmann.
- Numao, M.; Takagi, S.; and Nakamura, K. 2002. CAUI demonstration — composing music based on human feelings. In *Proc. AAAI 2002*. AAAI Press.
- Riecken, D. 1998. Wolfgang: “emotions” and architecture enable learning to compose music. SAB’98 Workshop on Grounding Emotions in Adaptive Systems. <http://www.ai.univie.ac.at/paolo/conf/sab98/sab98sub.html>.
- Rogers, S.; Fiechter, C.-N.; and Langley, P. 1999. An adaptive interactive agent for route advice. In *Proc. the Third International Conference on Autonomous Agents*, 198–205.
- Thom, B. 2000. Unsupervised learning and interactive Jazz/Blues improvisation. In *AAAI/IAAI*, 652–657.
- Tsunoda, K. 1996. Computer-supported composition of music. Master Thesis, University of Mie.
- Widmer, G. 1994. Learning with a qualitative domain theory by means of plausible explanations. In (*Michalski & Tecuci 1994*). chapter 25, 635–655.
- Wiggins, G., et al. 1999. Evolutionary methods for musical composition. *International Journal of Computing Anticipatory Systems*.