An intelligent hybrid model for chord prediction

URAQUITAN SIDNEY CUNHA and GEBER RAMALHO

Centro de Informática, Universidade Federal de Pernambuco, Cx. Postal 7851, 50740-540 Recife, PE, Brazil
E-mail: usgcc@di.ufpe.br glr@di.ufpe.br

For a system to be able to generate realtime accompaniment to previously unknown songs, it must predict their harmonic development, i.e. the chords to be played. We claim that such a system must combine long-term experience, to identify typical chord sequences (e.g. II–V and II–V–I), with 'on-the-fly' adaptation to track-recurrent structures (e.g. choruses and refrains) of the particular song being played. We have implemented a prediction system using a neural network model that encompasses prior knowledge about typical chord sequences. The results achieved are very encouraging, and rather better than those reported in the literature. However, our predictor could not adapt its behaviour to the idiosyncrasies of each song, since online learning is difficult in neural networks. In this paper, we propose an extension to our previous work by the inclusion of a rule-based sequence tracker, which detects recurrent chord sequences while the song is being performed. We show that this hybrid model, which combines a neural network predictor with a rule-based sequence tracker, improves the system's performance.

1. INTRODUCTION

The 1990s have witnessed the development of realtime interactive accompaniment systems, which can be used as rehearsal or concert partners, as well as arrangement assistants (Ramalho 1997, Ramalho, Rolland and Ganascia 1999). To improve the interactivity of these systems, it is important to add to them the capacity of predicting the next event (chords or notes) of the song being performed. This musical task is a particular case of time series prediction, which is concerned with foreseeing a new event based on a stream of events that have occurred so far (Weigend 1993). The success of time series prediction tasks depends on various issues, including problem dimensionality reduction, context representation, as well as knowledge representation and acquisition.

A central issue here is to build a predictor that can capture typical behaviour patterns of the problem class under study (prior knowledge), as well as the specific behaviour patterns of the particular problem instance (on-the-fly knowledge). This dichotomy between prior and on-the-fly knowledge is particularly important in the case of musical chord prediction. Prior knowledge of typical chord sequences (such as II–V–I, V–I, etc.) is of great help in prediction. However, it is also necessary to acquire on-the-fly knowledge concerning song-specific recurrent structures (such as choruses, sections, etc.). Even for experienced musicians, it can be very hard to perform a realtime accompaniment to an unknown song, mainly during its initial measures. However, as time goes by, musicians can detect some repetitive chord sequences and reuse them to improve their capacity to predict future chords.

From a computational standpoint, the problem is how to bring together the prior knowledge, learned earlier from several song examples, and the experience being acquired during the song execution.

We have implemented a prediction system using an MLP-backpropagation neural network (Rumelhart and McClelland 1986) encompassing prior knowledge about typical chord sequences (Cunha and Ramalho 1998). This knowledge is obtained by supervised learning (Mitchell 1997) on a corpus comprising more than thirty jazz standards. In our experiments, we have achieved good results (about 15% error rate). These are significantly better than those presented in the literature for tonal music realtime chord prediction (Thom and Dannemberg 1995). However, the difficulties of online learning in neural networks are well known. Without online learning, it would be impossible to adapt the behaviour of our neural network predictor according to the idiosyncrasies of each song. For this reason, we have proposed an extension of this neural network-based work to include a rule-based sequence tracker, which is capable of detecting recurrent chord sequences while the song is being performed. In this paper we present this hybrid model, showing that it improves the prediction rates with respect to the single neural network performance (Cunha 1999).

Sections 2 and 3 present, respectively, the chord prediction problem and the research efforts made to solve it. In section 4 we explain how our hybrid model was designed and developed. The results are

Organised Sound 4(2): 115–119

https://doi.org/10.1017/S1355771899002071 Published online by Cambridge University Press

Figure 1. Illustration of the chord prediction task.

presented in section 5. In section 6 we draw some final conclusions and point out future research goals.

2. THE PROBLEM

How can a computer predict, in real time, the chords of a previously unknown song? The chord predictor's task is to determine the 'next chord' of a song based only on the previously played chords and melody (see figure 1). The implementation of such a prediction system involves complex issues. First of all, chord prediction must be performed online, under hard time constraints. The realtime constraints have a great impact on the kind of algorithm that can be used, since the computational resources are limited.

Another issue is the fact that there are no universal rules for chord prediction. If chord chaining rules exist, they depend on various factors: the song, composer or arranger style, and the musician's current intentions (since this can change the song's harmony, e.g. by chord substitution). Moreover, each song has its own particularities, no matter what style, composer, arranger, etc. Despite the lack of universal prediction rules to explain the chord chaining of the whole song, some short chord sequences are recurrent (e.g. II–V–I, II–V). This fact motivated the use of different machine learning algorithms (Mitchell 1997) which can infer, from a given set of songs, rules for identifying these recurrent chord sequences (Thom and Dannemberg 1995).

The next issue to consider is the representation of musical context. In fact, even using a learning algorithm, it is necessary to determine how to represent the corpus. How many previously played chords must be considered to predict the current one? How do we represent the chord? Which attributes of a chord (root pitch, structure – major, minor, etc. – interval to previous chord, position within the song, etc.) are actually relevant to its definition? The success of the learning process strongly depends on an adequate choice of a representation of the chord (Cunha and Ramalho 1998).

The final issue is how to combine prior and on-the-fly knowledge, as discussed in section 1. Many peculiarities and recurrent structures of a song, such as refrain, chorus and stanzas, must be extracted in real time, since some important information is hidden within the global structure of each song. On the other hand, the general aspects of song structure identified by the machine learning process on the song examples cannot be neglected. There are several algorithms that can be used to learn prior knowledge from examples (Mitchell 1997). However, most of them cannot adapt the previously acquired knowledge to fit the structure of the song being played. For instance, in the chord sequence shown in figure 2, the chances of predicting that the chord B♭m7 would follow G6 are quite small, since it is an unusual modulation. Nevertheless, when this passage is repeated over and over, one can expect that musicians will play B♭m7 at the right moment.

3. STATE OF THE ART

Some research effort has been devoted to the prediction of musical parameters (Rowe 1993, Dirst and Weigend 1994, Hörnel and Ragg 1996). In the case of tonal music chord prediction, possibly the most relevant work was carried out at Carnegie Mellon University (CMU) (Thom and Dannemberg 1995). This used a learning algorithm based on *n-gram* models (Bell, Cleary and Witten 1990) to perform realtime chord prediction in a jazz songs context. The basic idea is to estimate, by training, the probability of occurrence of a chord, given its predecessors. Although the reported results were not satisfactory, Thom and Dannemberg discussed the chord prediction problem and proposed an elegant model for combining these two sources of knowledge: prior and on-the-fly.

Since the *n-gram* model technique provides both offline and incremental (online) learning, the implemented system bears three functioning modes. In the offline mode, the system uses the knowledge acquired by training on a set of about thirty songs to predict the chords of a unknown song. In the online mode, the system starts from scratch and acquires knowledge while trying to predict the chords of an unknown song. Finally, in the mix mode, both online and offline learning are combined. In this mode, the system starts with some initial knowledge and refines/extends it according to the particularities of the song whose chords are being predicted.

The tests done by the CMU team reached a precision rate of 42 to 53% on a corpus of about thirty songs, all with the same tonality. These unsatisfactory results may be due to the poor representation of the chords and the lack of information about melody (for a detailed discussion on this issue, see Cunha and Ramalho 1998, Cunha 1999). Despite the low prediction rate, the CMU team showed that the best results

Figure 2. Passage of *Darn That Dream* (by J. V. Heusen and E. DeLange).

were always achieved in the mix (online + offline) mode.

The model we proposed earlier was based on a neural network model and a richer representation of chords and songs (Cunha and Ramalho 1998). We used a multilayered perceptron model (MLP) with a backpropagation learning algorithm. The results we obtained, on a corpus of more than sixty jazz songs in different tonalities, were quite good: 88% of correct predictions on average.

This is a great improvement over the results of the CMU team. However, tests showed that the results would be even better if our model could adapt its behaviour on the fly. In fact, the nature of the neural network used was totally deterministic, in the sense that the same entry always produced the same output answers, and obviously, the same errors. In other words, it was not possible to undergo online adaptations.

This is due to the fact that it is hard to implement online adaptation of the learned connections weight. As we describe in the following section, we decided to include in our system a sequence tracker that could work in collaboration with the neural network. When the tracker is sure about what the next chord will be, it assumes control of the prediction system. In the other cases, the neural network performs the prediction.

4. THE CHORD PREDICTION MODEL

In this section we describe our hybrid model composed of the former neural net predictor (MLP-backpropagation) which learns and uses prior knowledge (Cunha and Ramalho 1998), and a sequence tracker, which analyses the structure of each song in real time to pursue the recurrent structures.

4.1. The sequence tracker

Jazz songs are formed from well-defined blocks that may be recurrent throughout the song, sometimes unchanged, sometimes with slight differences. These chord blocks form the structure of the jazz song and are informally called sections A, B, C or D (the most common structure being the 32AABA, composed of four 8-measure sections). For example, a song can begin with chord section A followed by section B. Next can be found another section with the same harmonic structure as A followed by another section B. A system capable of identifying when a block AB is to be repeated, for example, would not need to ask the machine learning module (i.e. the neural network) to try to find the right answer. It would simply follow the AB sequence already played in the song. Besides section repetitions, the song as a whole is usually repeated many times according, in general, to the following scheme: one theme exposition chorus, n improvisation choruses and one final theme exposition chorus. A chord predictor algorithm should also capture these repetitions.

From these principles, we have tried to define rules for realtime identification of these chord blocks within a song (including all of its repetitions). We have analysed the structure of about thirty jazz standards in order to extract sequence tracking rules, i.e. rules that determine when a given chord sequence is in fact a repetition of a previously played one. These rules guide the process of pattern matching, which is continuously done in order to detect relevant repetitions.

The sequence tracking process takes into account the melody as well as the chords. This is an essential point for the tracker's success and is also an innovation with respect to previous work, including our own.

The main sequence-tracking rules and principles are as follows:

(1) A sequence repetition is a block of a song including its melody and chords.
(2) The whole song is a sequence repetition.
(3) Each part of a song that has not yet been repeated within the same song is considered a sequence repetition candidate.
(4) One can only guarantee that a chord sequence is starting to be repeated after the exact match of at least three consecutive measures.
(5) Two consecutive chords or measures cannot be tested as belonging to different sequence repetitions because this can generate a loop.
(6) Usually, a sequence repetition has a quantity of measures that is a multiple of eight.
(7) In many cases, sequences within a song are not completely repeated, presenting some differences in measures $8n$ or $8n+1$, where $n = 1, 2, 3 \ldots$. In this case, the sequence tracker passes the control to the neural network.
(8) Each sequence must be tested until the beginning of the next sequence is found, to avoid loops.

1	Bb	Cm7	C#dim	Bb	Bb	Cm7	C#dim	Bb
3	Bb	Dbm6	Cm7	F9	Bb	Dbm6	Cm7	F9
5	Bb		Bb7		Eb6		Ebm6	
7	Bb				Bb		F7	
9	Bb	Cm7	C#dim	Bb	Bb	Cm7	C#dim	Bb
11	Bb	Dbm6	Cm7	F9	Bb	Dbm6	Cm7	F9
13	Bb		Bb7		Eb6		Ebm6	
15	Bb				Bb			

Figure 3. Chord grid of *Basin Street Blues* (by S. Williams).

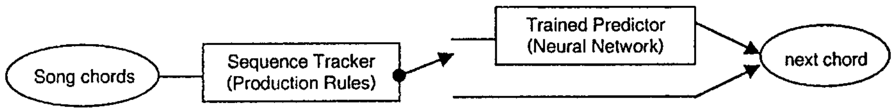

Figure 4. Hybrid model scheme.

Figure 3 shows an example of the *Basin Street Blues* chord grid. Because of rule 5, the tracker will not try to interpret measure 2 (or measure 10) as the repetition of measure 1 (or measure 9). On the other hand, the tracker will suspect that the sequence beginning at measure 9 is the repetition of that starting at the first measure (if the respective melodies also match – see rule 1). From measure 12 (rule 4), the tracker is sure that there is such a repetition and then assumes control of the whole prediction system, repeating the chords of measures 1–8. However, according to rule 7, the tracker will not try to play the chords (B♭, F7) of the final measure of the tracked sequence. Then, at measure 16, control returns to the neural network predictor.

4.2. The hybrid model

The hybrid model we propose combines a neural network predictor and a rule-based sequence tracker (figure 4). The sequence tracker monitors the chord stream. When a sequence repetition is detected, the tracker assumes control of the whole prediction system, indicating what the next chord will be. When the tracker is not sure that a sequence is being repeated, or when the tracked sequence finishes, it returns control to the neural network. Thus, the tracker is only used when the prediction is 100% guaranteed. Otherwise, it is preferable to rely on the neural network predictions, which are near to 90% correct.

5. IMPLEMENTATION AND RESULTS

The system was developed in Borland Delphi 3.0 (Delphi 1997). The sequence tracking rules were translated into commands in procedural form. The neural network was tested with Qnet97 software (Qnet97 1997) and implemented in Delphi. When the tests were concluded, we created an integrated environment for chord prediction for the Windows 95/98 platform. With a user-friendly interface, the software provides visualisation of the whole process of prediction in real time (figure 5).

Experimental evaluations have been undertaken with about sixty jazz standards in different tonalities. The results obtained with the hybrid model are better than those using just a neural network to solve the problem, as shown in the table. Furthermore, the tests have shown that the more the song is repeated, the lower is the error rate. These results show that this model can capture, on the fly, new knowledge in the process.

6. CONCLUSION

We have proposed an original model for chord prediction which combines prior knowledge with online adaptation. The results have encouraged us to develop chord prediction systems beyond the jazz style, which will be the basis of realtime accompaniment systems.

Table. Comparison of results usng the hybrid model.

Error rate with neural network only	Error rate of the hybrid system (song played once)	Error rate of the hybrid system (song played twice)	Error rate of the hybrid system (song played three times)
12%	8.8%	4.1%	0.0%

https://doi.org/10.1017/S1355771899002071 Published online by Cambridge University Press

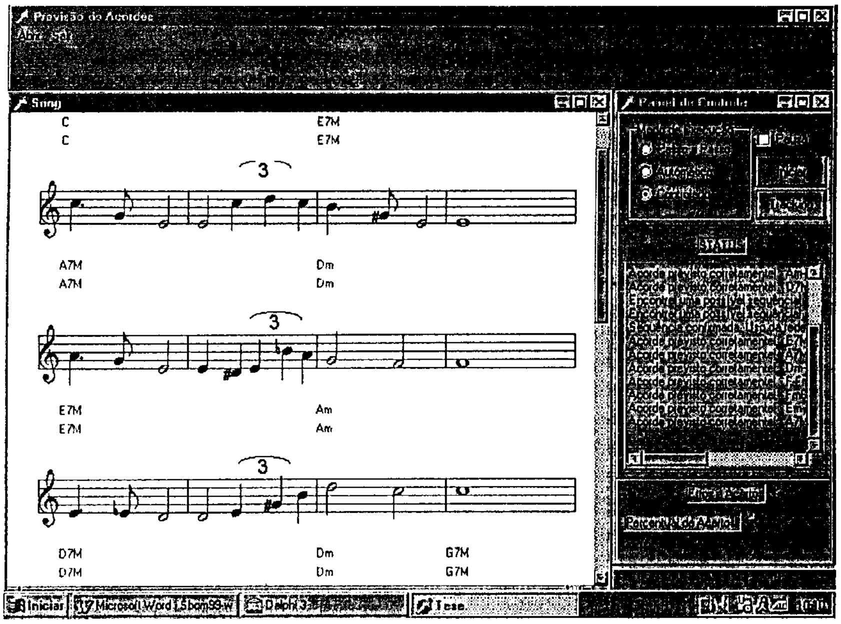

Figure 5. Main window of our prediction system.

In spite of the good results, we are aware of the complexity of this problem and of the necessity for new and more detailed analysis of the best ways to proceed. We intend to continue our research and to extend our model to the prediction of other musical parameters.

REFERENCES

Bell, T., Cleary, J., and Witten, I. 1990. *Text Compression*. Prentice Hall Advanced Reference Series. New Jersey: Prentice Hall.

Cunha, U. 1999. *Um ambiente hibrido inteligente para previsão de acordes musicais em tempo real*. MSc Dissertation, Universidade Federal de Pernambuco.

Cunha, U., and Ramalho, G. 1998. Previsao de acordes em musicas tonais. *Proc. of the 5th Brazilian Symp. on Computer Music*, pp. 45–54. SBC&M'98. Belo Horizonte: SBC editions.

Delphi. 1997. *Delphi 3.0.* Inprise Corporation, 100 Enterprise Way, Scotts Valley, CA 95066–3249, USA. Available at http://www.borland.com/delphi

Dirst, M., and Weigend, A. S. 1994. Baroque forecasting: On completing J. S. Bach's last fugue. In A. S. Weigend and N. A. Gershenfeld (eds.) *Time Series Prediction: Forecasting the Future and Understanding the Past*, pp. 151–72. Reading, MA: Addison-Wesley.

Hörnel, D., and Ragg, T. 1996. Learning musical structure and style by recognition, prediction and evolution. In D. Rossiter (ed.) *Proc. of the 1996 Int. Computer Music Conf.*, pp. 59–62. San Francisco: International Computer Music Association.

Mitchell, T. 1997. *Machine Learning*. New York: McGraw-Hill.

Mozer, M. C. 1994. Neural network music composition by prediction: Exploring the benefits of psychophysical constraints and multiscale processing. *Connection Science* **6**: 247–80.

Qnet97. 1997. *Qnet97*. Vesta Services, Inc. 1001 Green Bay Rd, STE196 Winnetka, IL 60093, USA. Available at http://www.qnet97.com

Ramalho, G. 1997. Pattern reuse in tonal music improvisation and accompaniment systems. *Proc. of the 4th Brazilian Symp. On Computer Music*, pp. 47–58. SBC&M'97. Brasilia: SBC editions.

Ramalho, G., Rolland, P.-Y., and Ganascia, J.-G. 1999. An artificially intelligent jazz performer. *Journal of New Music Research* **28**(2).

Rowe, R. 1993. *Interactive Music Systems*. Cambridge, MA: MIT Press.

Rumelhart, D., and McClelland, L. 1986. *Parallel Distributed Processing*, Vol. 1. Cambridge, MA: MIT Press.

Russel, S., and Norvig, P. 1995. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs: Prentice Hall.

Thom, B., and Dannemberg, R. 1995. Predicting chordal transitions in jazz: The good, the bad and the ugly. VI Workshop on Music and AI. *14th Int. Joint Conf. on Artificial Intelligence (IJCAI–95)*. Montréal, Canada.

Weigend, A. 1993. *Time Series Prediction: Forecasting the Future and Understanding the Past*. London: Addison Wesley.

https://doi.org/10.1017/S1355771899002071 Published online by Cambridge University Press