Query Parsing Using Probabilistic Tree Grammars

José F. Bernabeu JFBERNABEU@DLSI.UA.ES
Jorge Calera-Rubio CALERA@DLSI.UA.ES
José M. Iñesta INESTA@DLSI.UA.ES
David Rizo DRIZO@DLSI.UA.ES
Dept. Lenguajes y Sistemas Informáticos, Universidad de Alicante, Spain

Abstract

The tree representation, using rhythm for defining the tree structure and pitch information for node labeling has proven to be effective in melodic similarity computation. In this paper we propose a solution representing melodies by tree grammars. For that, we infer a probabilistic context-free grammars for the melodies in a database, using their tree coding (with duration and pitch) and classify queries represented as a string of pitches. We aim to assess their ability to identify a noisy snippet query among a set of songs stored in symbolic format.

1. Introduction

Symbolic music retrieval from queries has been extensively studied in the MIR literature. In order to search a query in a dataset of songs, we can apply any of the well-known pattern matching algorithms like local string editing to each of the songs in the dataset, and retrieve a ranking of most similar items. The main problem here is the efficiency when using large datasets, but with the advantage that it can find a partial or approximate occurrence of the query in any part of the dataset. On the other hand, a previous indexation of the dataset, e.g by means of motive extraction, solves the scalability issue, but motive extraction usually needs to work with exact repetitions, making very inaccurate to build robust indices from different interpretations of the same song. In addition, when searching only in motives, the music not present in the motives is hidden.

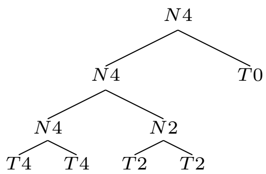
Our proposal is able to solve intrinsically these problems. The probabilistic grammar structure itself encodes both motives and melody variations, giving more weight to the most repeated themes, without the need to be exact, and enabling the possibility to learn from different renderings of the same song, leaving aside the need to encode motives. Besides, there is no difference between looking for a whole song or searching a small query in the dataset. For it, we use probabilistic context-free grammars obtained from probabilistic k-testable tree models (Knuutila, 1993). The duration information implicit in the tree representation is captured by the grammar and this is used for classifying snippet melodies represented by strings that only have the pitch information. This is a solution when duration information is not available or unreliable for the input data. This work is an extension of that by Bernabeu et al. (Bernabeu et al., 2011) where the inputs were whole melodies.

2. Methodology

For representing the note pitches in a monophonic melody s as a string, symbols σ from a pitch representation alphabet Σ_p are used: $s \in \Sigma_p^*, s = \sigma_1\sigma_2...\sigma_{|s|}$. In this paper, the interval from the tonic of the song modulo 12 is utilized as pitch descriptor and the symbol '$-$' to represent rests. In the tree approach, each melody bar is represented by a tree, $t \in T_{\Sigma_p}$. Bars are coded by separated trees and then they are linked to a common root. The level of a node in the tree determines the duration it represents. Once the tree has been built, a bottom-up propagation of the pitch labels is performed to label all the internal nodes.

Stochastic models based on k-grams predict the probability of the next symbol in a sequence depending on the $k-1$ previous symbols (Downie, 1999). k-gram models can be regarded as a probabilistic extension of locally testable languages. These models are easy to learn and can be efficiently processed. Locally testable languages, in the case of trees, were described

Appearing in *Proceedings of the 29th International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012.



(1/1) $I \rightarrow N4$
(1/3) $N4 \rightarrow N4 \ \ T0$
(1/3) $N4 \rightarrow N4 \ \ N2$
(1/3) $N4 \rightarrow T4 \ \ T4$
(1/1) $N2 \rightarrow T2 \ \ T2$

Figure 1. Example of a probabilistic tree grammar ($k = 2$).

by Knuutila (Knuutila, 1993). These kind of tree languages can be defined using the formalism of deterministic tree automata (DTA) and can be extended to the case where the sample Ω is stochastically generated, incorporating probabilities to the DTA (PDTA).

We can classify a new melody in a particular class. For that, we need to infer a PDTA for each class, C_j, from well classified melodies. Once the PDTAs for the different classes have been inferred and the probabilities estimated, a melody M can be classified in the class $\hat{C}$ that maximizes the likelihood. In order to do this, both training and new melodies must be represented by trees. However, in a practical situation, the user of a music retrieval system may make a query using just a sequence of pitches or with non reliable durations. Therefore, we need to transform the k-testable tree automata in probabilistic context-free grammars to use them for parsing the input melody strings. Thus, we could use these grammars to obtain the correct structure for a given melody represented by a pitch string.

A probabilistic context-free grammar (PCFG) is shown in Figure 1 (right) corresponding to the tree in the left. Therefore, we can obtain k-testable grammars with different values for k (depth of subtrees, see (Bernabeu et al., 2011) for details).

In general, k-testable grammars with larger values of k contain more specialized rules and, therefore, they are less ambiguous and allow for faster parsing. But the larger they are, the more likely they can find unseen strings, that will be assigned a zero probability in recognition. Therefore, the use of smoothing techniques becomes necessary if one wants to use these models for parsing using the proper context size. Here, we use a rule-based backing-off approach. This requires the implementation of specific parsers, since building the whole grammar is unfeasible due to the large number of implicit rules. An alternative scheme that requires only minor modifications is to use a quasi-equivalent grammar G' where the different k-grammars are combined to build a unique grammar for each class C_j.

After a grammar G_j is inferred for each class C_j we need an algorithm for obtaining the probability that a given string s is generated by a grammar G_j. For this purpose, we have used the Stolcke algorithm and the CYK+ algorithm for string parsing. These parsing algorithms are able to give the probability $p(s|G)$ that a string s is generated by a probabilistic Context-free grammar G without requiring conversions to Chomsky Normal Form (CNF). Then, a melody query M is classified in the class $\hat{C}$ that maximizes the likelihood

$$\hat{C} = \arg\max_{j} \ l(M|C_j) \tag{1}$$

3. Results and conclusions

In this work, we want to assess the ability of the tree grammars to identify a noisy snippet query among a set of songs stored in symbolic format. For that, we use a corpus consisting of a set of 420 monophonic 8-12 bar incipits of 20 worldwide well known tunes of different musical genres. Then, different $(3-22)$ length queries have been built from tunes using only the pitch data. Then, a tree grammar for each tune has been trained from the tunes represented by trees. Finally, the queries have been classified in the grammar whose probability to generate the query is higher.

The obtained results show the system obtain a success rate of about 73% for 18 query length. Moreover, the system reaches a success rate of about 95% when a ranking is formed with the best four classes.

Thus, it seems that tree grammars are able to classify noisy snippet queries. Although the success rate seems to increase with the query length we need find a balance between success and computation time. Moreover, the ranking could be used as a pruning for discarding the parse of a whole melody for the worse grammars given the snippet query. In other words, if we have a long melody, we could classify it using only its beginning. After that, we compute the probability of the whole melody only for the best classes. This way, we can improve the computing time when the melody length is large.

References

Bernabeu, J. F., Calera-Rubio, J., and Iñesta, J. M. Classifying melodies using tree grammars. volume 6669, pp. 572–579, 2011.

Downie, J. Stephen. *Evaluating a Simple Approach to Music Information Retrieval: Conceiving Melodic n-grams as Text.* PhD thesis, University of Western Ontario, 1999.

Knuutila, T. Inference of k-testable tree languages. In *Bunke (Ed.), Advances in Structural and Syntactic Pattern Recognition (Proc. of the S+SSPR'92), World Scientific, Singapore*, 1993.