Proceedings 1st International Workshop on Advanced Analytics and Learning on Temporal Data
AALTD 2015

Sequential pattern mining on multimedia data

Corentin Hardy, Laurent Amsaleg, Guillaume Gravier, Simon Malinowski and René Quiniou
PIRISA/Inria Rennes, France

Abstract. Analyzing multimedia data is a challenging problem due to the quantity and complexity of such data. Mining for frequently recurring patterns is a task often ran to help discovering the underlying structure hidden in the data. In this article, we propose audio data symbolization and sequential pattern mining methods to extract patterns from audio streams. Experiments show that this task is hard and that the symbolization is a critical step for extracting relevant audio patterns.

1 Introduction

The amount of multimedia data grows from day to day with ever increasing acquisition and storage capabilities. In turn, analyzing such complex data to extract knowledge is a challenging problem. For instance, analysts are looking for methods that could help to discover the underlying structure of multimedia documents such as video or audio streams. Unsupervised extraction of recurrent patterns and finding their occurrences in the data could provide such a segmentation and could achieve a first step towards the automatic understanding of multimedia data. In an audio stream, a word, a jingle, or an advertisement could typically represent a pattern. However, the variability of audio motifs makes pattern mining difficult, especially audio motifs related to words, since the variability due to different speakers and channels is high.

Overall, the extraction of repeated motifs in time series is a very active domain. Two kinds of approaches have been proposed: the first one consists in working directly with the time series and in finding close sub-sequences based on a distance measure such as the Euclidean or the Dynamic Time Warping (DTW) [1] distances. The second one consists in transforming the time series into sequences of symbols to then use sequential motif discovery algorithms [2]. Very few works have investigated the second approach; this preliminary work thus explores how to use sequential pattern mining algorithms on audio data.

This paper is organized as follows. In Section 2, we review the related work about motif discovery in audio data. In Section 3, we explain our proposed approach. Section 4 presents preliminary results and section 5 concludes and discusses future issues for this work.

2 Related work

Motif discovery relies either on raw time series processing or on mining a symbolic version [3,4,5]. In the first kind of approaches, algorithms are mostly built on

the DTW distance which can deal with temporal distortions that often occurs in audio signals [6]. Muscariello et al. [7] have proposed an extended version of the DTW for finding the best occurrence of a seed in a longer subsequence. This kind of approaches is efficient in terms of accuracy as the signal is completely exploited but the computational cost of the DTW distance prevents its use on very large databases.

Other approaches working with a symbolized version of the audio signal mostly use algorithms from bioinformatics to extract motifs. In [8], the MEME algorithm [9] is used to estimate a statistical model for each discovered motif. In [10], the SNAP algorithm [11] is used to search by query near-duplicate video sequences.

Some algorithms coming from bioinformatics are very efficient, but have been optimized to work with alphabets of very small size (from 4 to 20). In this paper, we consider the use of sequential pattern mining algorithms for discovering motifs in audio data.

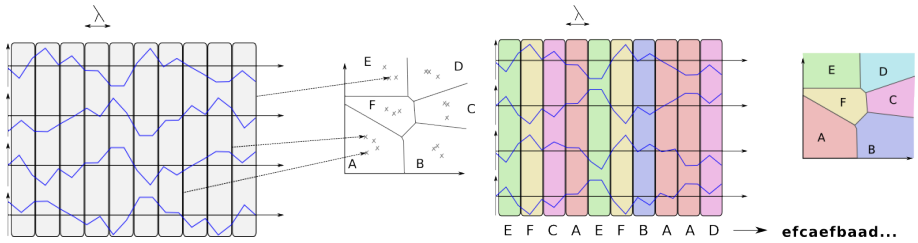
3 Pattern mining on audio data

In this section, we explain how we used sequential pattern mining algorithms to discover repeating patterns in audio data. As pattern mining algorithms deal with symbolic sequences, we present first how to transform time series related to audio data into symbolic sequences. Then we show how to use sequential pattern algorithms on symbolic sequences.

MFCC (Mel-frequency cepstral coefficients) is a popular method for representing audio signals. First, MFCC coefficients are extracted from the raw audio signal (with a sliding window) yielding a 13-dimensional time series. Then, this multivariate time series is transformed into a sequence of symbols. Many methods have been proposed for transforming time series into a sequence of symbols. Here, we have chosen to use a method proposed by Wang et al. [12]. We have also tried the very popular SAX approach [2]. SAX symbols contain very few information about the original signal (only the average value on a window). This symbolisation technique is less adapted to our problem and produced worse results.

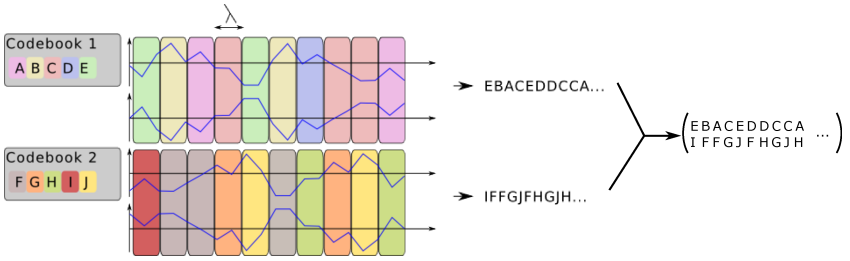
To this end, each dimension of the 13-dimensional time series is divided into consecutive non-overlapping windows of length λ. The 13 sub-series related to the same window are then concatenated (respecting the order of the MFCC data). The resulting vectors of size $13 \times \lambda$ are then clustered by a k-means algorithm for building a codebook, each word in the codebook corresponding to a cluster. Finally, the original multivariate time series is coded into a sequence of symbols by assigning to each window the symbol in the codebook corresponding to the closest cluster centroid. This symbolization process is sketched in Figures 1a and 1b.

The representation above could be too imprecise as it mixes coefficients of very different order. To cope with this problem we propose to divide the 13 dimensions into 2 or more sub-bands of consecutive dimensions that represent



(a) K-means clustering is performed on the set of windows to build a codebook (of size 6, here).

(b) Every window is labeled by the symbol associated with the closest cluster centroid.



(c) Conversion of a 2 sub-band times series into a sequence of itemsets using 2 codebook of size 5.

Fig. 1: Time series symbolization into a sequence of items (figures 1a and 1b) and a sequence of itemsets (figure 1c).

more closely related dimensions. The same transformation described above operates on sub-bands and yields one codebook per sub-band. There are thus as many symbolic sequences as there are sub-bands. Finally, the sub-band symbols related to the same windows are grouped into itemsets in the Figure 1c.

Once the raw signal is transformed into a symbolic sequence of items or itemsets, classical sequential motif discovery algorithms can be applied. Two kinds of sequential pattern discovery algorithms have been proposed: algorithms that process sequences of items and algorithms that process sequences of itemsets (an itemset is a set of items that occur in a short time period). We have chosen to evaluate one algorithm of each kind in this paper: MaxMotif [13] and CMP-Miner [14] that process respectively sequences of items and sequences of itemsets.

Note that, in the classical setting of sequential pattern mining, a pattern occurrence may skip symbols in the sequence. For instance, ***acccb*** is an occurrence of pattern *ab* in sequence *dacccbe*. Generally, algorithms provide means to put constraints on extracted motifs, such as minimum and maximal motif length and the allowed gaps; gaps are symbols that can be skipped when looking for a pattern occurrence. In our application, it is crucial to allow gaps in motifs since temporal distortions often occurs in audio signals.

MaxMotif enumerates all frequent (with respect to a given minimal support) closed patterns in a database of item sequences. MaxMotif allows gaps in the

temporal domain (represented by the *wildcard* symbol $-$). For instance, pattern $(f - a)$ occurs in sequence (e**fca**e**fba**ab) at positions 2 and 6.

CMP-Miner extracts all frequent closed patterns in a database of itemset sequences. It uses the PrefixSpan projection principle [15] and the BIDE bidirectional checking [16]. CMP-Miner allows gaps both in the temporal domain and inside an itemset. For instance, pattern $\begin{pmatrix} b & - & c & - \\ f & - & g & j \end{pmatrix}$ occurs in sequence $\begin{pmatrix} e & b & a & c & e & b & d & c & c & a \\ i & f & f & g & j & f & h & g & j & h \end{pmatrix}$ at positions 2 and 6.

The parameters of the two methods are described in Table 1.

Table 1: List of parameters

Methods	**Symbolization**	**Parameters for mining**
MaxMotif	α, size of codebook.	*minSupport*, minimal support.
	λ, length of windows.	*maxGap*, maximal gap between 2 consecutive items in a pattern.
		maxLength, maximal pattern length.
		minLength, minimal pattern length.
CMP-Miner	α, size of codebook.	*minSupport*, minimal support.
	λ, length of windows.	*maxGap*, maximal gap between 2 consecutive itemsets in a pattern.
	β, number of bands.	*minItem*, minimal number of items in itemsets.
		maxLength, maximal pattern length.
		minLength, minimal pattern length.

4 Experiments

We present in this section some results from two experiments, one on a synthetic dataset and the other on a real dataset.

4.1 Experiment on a synthetic dataset

In this first experiment, we have created a dataset composed of 30 audio signals corresponding to 10 utterances of the 3 words "affaires", "mondiale" and "cinquante" pronounced by several French speakers. Our goal is to evaluate the impact of the codebook size on the extracted motifs. The two algorithms presented above have been applied on this dataset with the following parameters: $\lambda = 5$, $minSupport = 4$, $maxGap = 1$, $minLength = 4$, $maxLength = 20$. For CMP-Miner we set $\beta = 3$ and $minItem = 2$. These parameter settings were chosen after extensive tests on possible value ranges.

First, sequential patterns are extracted. Then, we associate with each pattern the word in the utterances of which this pattern most often occurs. For each

extracted pattern, a precision/recall score is computed. Figure 2a and 2b depict the precision/recall score versus the codebook size for MaxMotif and CMP-Miner. As can be seen, MaxMotif obtains the best efficiency. This figure also shows that when the codebook size increases, the precision improves slightly but not the recall.

Figure 2c shows the pattern length distribution for different codebook sizes for MaxMotif. For small codebooks, many long patterns are extracted. However, they are not very accurate because, being general, they can occur in many different sequences. For big codebooks, many pattern candidates can be found, reflecting sequence variability. However, many candidates have a low support, often under the minimal threshold, and, so, less patterns are extracted.

The symbolization step is crucial. Figure 2d shows five symbolic representations of the word "cinquante" for a codebook of size 15. These strings highlight the two kinds of variability (spectral and temporal) that makes the task hard for mining algorithms in this example. The same experiment was performed using the SAX symbolization method [2] on each dimension of the multidimensional times series. This representation revealed to be less accurate. Indeed, the results obtained by CMP-Miner using the SAX representation were worse. There is no space to detail these results here.

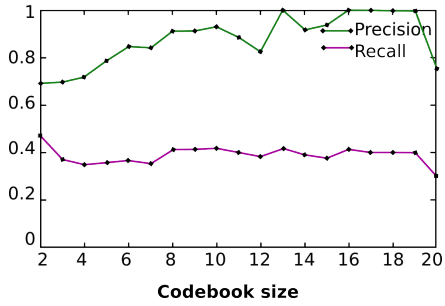
4.2 Experiment on a larger database

Now, we consider a dataset containing 7 hours of audio content. The dataset is divided into 21 audio tracks coming from various radio stations. This experience is closer to a real setting.

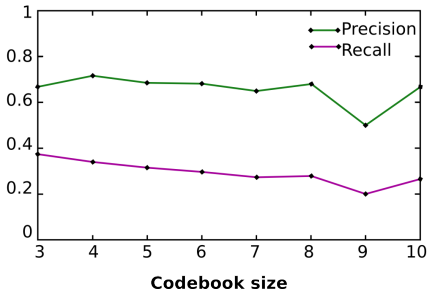
Only MaxMotif has been tested on this dataset. The parameters were: $\lambda = 4$, $\alpha = 80$, $minSupport = 40$, $maxGap = 1$, $minLength = 5$, $maxLength = 20$. The codebook size is greater than in the previous experiment to deal with more different sounds. Pattern extraction is very fast: less than 4 minutes for more than one million of patterns. Some of them are interesting and correspond, for instance, to crowd noises, jingle and music patterns or short silence. However, similarly to the experiment on the synthetic dataset, only very few patterns corresponding to repeated words could be extracted.

5 Conclusion

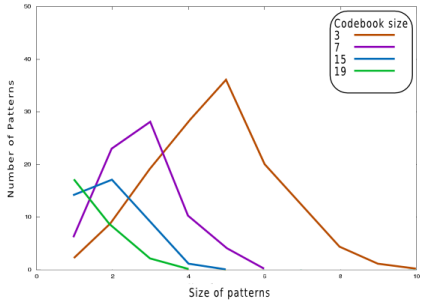
In this paper, we have presented a preliminary work investigating how to use sequential pattern mining algorithms for audio data. The aim of this work was to evaluate whether these algorithms could be relevant for this problem. The experiments pointed out the difficulty to mine audio signals, because of temporal and spectral distortion. Same words pronounced in different contexts and by different speakers can be very different and yield very different patterns. The results are promising but both symbolization and motif extraction should be improved. For instance, to account for spectral variability, considering distances between symbols should improve the overall performance of pattern extraction.



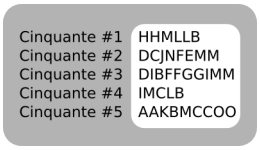
(a) Precision/Recall curves for MaxMotif.



(b) Precision/Recall curves for CMP-Miner with 3 sub-bands.



(c) Pattern size distribution for different size of codebook.



(d) Example of representation for a codebook of size 15.

Fig. 2: Results of experience on synthetic data.

We have also noticed that all the dimensions of the MFCC times series are not as important for the discovery. Selecting or weighting the dimensions of multidimensional time series could improve the performance too.

References

1. A. Mueen, "Enumeration of time series motifs of all lengths," in *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pp. 547–556, Dec 2013.
2. J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing sax: A novel symbolic representation of time series," *Data Min. Knowl. Discov.*, vol. 15, pp. 107–144, Oct. 2007.
3. C. Herley, "ARGOS: Automatically extracting repeating objects from multimedia streams," *IEEE Trans. on Multimedia*, vol. 8, pp. 115–129, Feb. 2006.
4. P. Esling and C. Agon, "Time-series data mining," *ACM Computing Surveys (CSUR)*, vol. 45, no. 1, p. 12, 2012.
5. C. H. Mooney and J. F. Roddick, "Sequential pattern mining – approaches and algorithms," *ACM Comput. Surv.*, vol. 45, Mar. 2013.

6. A. Park and J. R. Glass, "Unsupervised pattern discovery in speech," *IEEE Transaction on Acoustic, Speech and Language Processing*, vol. 16, pp. 186–197, Jan. 2008.
7. A. Muscariello, G. Gravier, and F. Bimbot, "Audio keyword extraction by unsupervised word discovery," in *INTERSPEECH 2009: 10th Annual Conference of the International Speech Communication Association*, (Brighton, United Kingdom), Sept. 2009.
8. J. J. Burred, "Genetic motif discovery applied to audio analysis," in *International Conference on Acoustics, Speech and Signal Processing*, pp. 361–364, IEEE, 2012.
9. T. Bailey and C. Elkan, "Unsupervised learning of multiple motifs in biopolymers using expectation maximization," *Machine Learning*, vol. 21, no. 1-2, pp. 51–80, 1995.
10. L. S. d. Oliveira, Z. K. do Patrocinio, S. J. F. Guimarães, and G. Gravier, "Searching for near-duplicate video sequences from a scalable sequence aligner," in *International Symposium on Multimedia*, pp. 223–226, IEEE, 2013.
11. M. Zaharia, W. J. Bolosky, K. Curtis, A. Fox, D. Patterson, S. Shenker, I. Stoica, R. M. Karp, and T. Sittler, "Faster and more accurate sequence alignment with snap," *arXiv preprint arXiv:1111.5572*, 2011.
12. Q. Wang, V. Megalooikonomou, and C. Faloutsos, "Time series analysis with multiple resolutions," *Information Systems*, vol. 35, no. 1, pp. 56–74, 2010.
13. H. Arimura and T. Uno, "An efficient polynomial space and polynomial delay algorithm for enumeration of maximal motifs in a sequence," *Journal of Combinatorial Optimization*, vol. 13, 2007.
14. A. J. Lee, H.-W. Wu, T.-Y. Lee, Y.-H. Liu, and K.-T. Chen, "Mining closed patterns in multi-sequence time-series databases," *Data & Knowledge Engineering*, vol. 68, no. 10, pp. 1071 – 1090, 2009.
15. J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M.-C. Hsu, "Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth," in *International Conference on Data Engineering*, p. 0215, 2001.
16. J. Wang and J. Han, "Bide: efficient mining of frequent closed sequences," in *Data Engineering, Proceedings. 20th International Conference on Data Engineering*, pp. 79–90, March 2004.