UNSUPERVISED ALGORITHM FOR RETRIEVING CHARACTERISTIC PATTERNS FROM TIME-WARPED DATA COLLECTIONS

Tomáš Kocyan, Jan Martinovič, Michal Podhorányi, Ivo Vondrák

VŠB - Technical University of Ostrava
IT4Innovations
17. listopadu 15/2172, 708 33 Ostrava, Czech Republic

tomas.kocyan@vsb.cz, jan.martinovic@vsb.cz, michal.podhoranyi@vsb.cz, ivo.vondrak@vsb.cz

ABSTRACT
This paper discusses possibilities of using the Voting Experts algorithm enhanced by the Dynamic Time Warping (DTW) method for improving performance of Case-Based Reasoning (CBR) methodology used with time-warped data collections. CBR, in general, is the process of solving new problems based on the solutions of similar past problems. Success of this methodology strongly depends on the ability to find similar past situations. Searching these similar situations in data collections with components generated in equidistant time and in finite number of levels is now a trivial task. The problem arises for data collections that are subject to different types of distortions (e.g. measurement of natural phenomena such as precipitations, measured discharge volume etc.). The main goal of this paper is to provide suitable mechanism for retrieving typical patterns from distorted time series and thus improve the usability of CBR.

Keywords: segmenting, case-based reasoning, voting experts, dynamic time warping, time series

1. INTRODUCTION

Many types of existing collections often contain repeating sequences which could be called as patterns (Theodoridis and Koutroumbas 2006). If these patterns are recognized they can be for instance used in data compression or for prediction. Extraction of these patterns from data collections with components generated in equidistant time and in finite number of levels is now a trivial task. The problem arises for data collections that are subject to different types of distortions in all axes. This paper will focus on processing of measured river discharge volume, especially on finding typical patterns in this data collection. In our future research, such found patterns will be used for simulation of the rainfall runoff process and for prediction of discharge volumes in basin's outlet cross section.

River discharge is an important value and is frequently monitored along many major rivers and streams in the world. River discharge is the amount of water that flows through the river bed. There are mathematical formulas to measure the volume of water that flows through a certain section of the river at a given time. River discharge is very variable value. The existence of an increasing or decreasing trend in a river discharge time series data can be induced by the change in the climatic factors such as temperature (e.g. Comani 1987) or precipitation (Maheras and Kolyva-Machera 1990). Variability in river runoff has been reported in, for example, Giakoumakis and Baloutsos (1997). For this reason, it is almost impossible to use conventional methods for patterns extraction and it is necessary to use an alternative approach (Fanta, Zaake and Kachroo 2001). Found patterns will be presented in river hydrograph, which is a graph that shows the change in river discharge over time. Different river catchments produce different shapes of hydrograph (river discharge variability).

2. CASE-BASED REASONING

Case-Based Reasoning methodology belongs to a group of artificial intelligence methods and it can be simply described as a process of solving new problems based on the solutions of similar past problems. Each of these cases consists of a specific problem, its solution, and the way it was achieved. For purposes of computer reasoning the CBR has been formalized as a four-step abstract process (Watson 1997):

- Retrieve - Retrieve the most similar cases from case database that are relevant to solving it.
- Reuse - Map the found solutions from the previous cases to the new problem.
- Revise - Test the derived solution in the real conditions and, if necessary, revise it.
- Retain - After the solution has been successfully adapted to the target problem, the resulting experience is stored as a new case in memory and can be used for prediction in the next cycle.

For achieving the best results using Case-Based Reasoning methodology, it is necessary to successfully manage the first step (Retrieve). Many supervised and unsupervised methods for looking for patterns and similar situations exist, but the most of them have a common problem: they cannot handle searching for patterns of different lengths and they are not resistant to distortion. The Voting Experts algorithm is one of the algorithms that are able to handle these problems.

3. VOTING EXPERTS

The Voting Expert Algorithm is a domain-independent unsupervised algorithm for segmenting categorical time series into meaningful episodes. It was first presented by Cohen and Adams (2001). Since this introduction, the algorithm has been extended and improved in many ways, but the main idea is always the same. The basic Voting Experts algorithm is based on the simple hypothesis that natural breaks in a sequence are usually accompanied by two statistical indicators (Cohen, Adams, and Heeringa 2007): low internal entropy of episode and high boundary entropy between episodes.

The basic Voting Experts algorithm consists of the following three main steps:

- Build an nGram tree from the input, calculate statistics for each node of this tree (internal and boundary entropy) and standardize these values in nodes at the same depth.
- Pass a sliding window of length *n* over the input and let experts vote. Each of the experts has its own point of view on current context (current content of the sliding window) and votes for the best location for the split. The first expert votes for locations with the highest boundary entropy, the second expert votes for locations with a minimal sum of internal split entropy. By this way, the votes are counted for each location in the input.
- Look for local maximums which overcome selected threshold. These points are adepts for a split of sequence.

For detailed explanation of each of mentioned steps see (Cohen, Adams, and Heeringa 2007). Tests showed that the algorithm is able to segment selected input into meaningful episodes successfully. It was tested in many domains of interest, such as looking for words in a text (Cohen and Adams 2001) or segmenting of speech record (Miller, Wong, and Stoytchev 2009).

There are several ways how to improve the basic Voting Experts algorithm. Simply we can divide these improvements into the two main groups. On the one hand, a custom "expert" can be added to voting process (for example Markov Expert by Cheng and Mitzenmacher (2005)) and receive additional point of view on your input. On the other hand, there are methods based on repeated or hierarchical segmenting of the input (Miller and Stoytchev 2008, Hewlett and Cohen 2009).

One of the simplest ways how to slightly improve performance of segmenting is two-way passing of the sliding window. It means using classic voting algorithm supplemented by segmenting of reversed input. This idea was outlined in (Hewlett and Cohen 2009) which showed the way to make high-precision cut points by selection of higher values of the threshold. Additionally, reversing the corpus and segmenting the reversed input with Voting Experts generates a different set of backward cut points. The subsequent intersection of sets of cut points offers high precision segmenting. However, on the other hand, this high precision causes loss of recall.

4. VOTING EXPERTS POST-PROCESS

Proposed solution for Voting Experts improvement takes the task of using Dynamic Time Warping algorithm (introduced below) and high precision cuts as a starting point for looking for typical patterns located in the input. The basic idea is to refine the sparse set of high precision cuts into regular sequences as correctly as possible. The mentioned refinement will be done by several types of post-processing methods and the results will be compared. Methods will differ, but they share a common principle (as shown in Figure 1). If there are high precision cuts in the input (such as cuts A, B, C and D in Figure 1) and if the shorter sequence (bounded by cuts C and D) is a subsequence of the longer one (bounded by cuts A and B), we can deduce new boundaries E and F by projecting the boundaries of common subsequence to the longer sequence. In this very simplified example the sequences were composed by definite number of values and limited length, so the evaluation is quite straightforward.

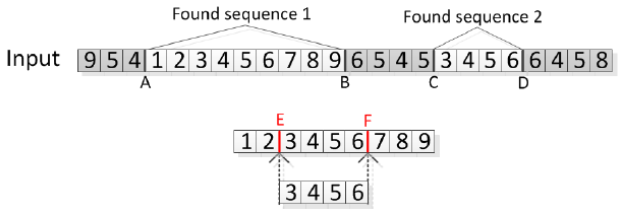

Figure 1: Refinement of sequences

In the case of application of previously mentioned process on distorted data, it is necessary to slightly modify it. Typical episodes of measurement of natural phenomena (such as precipitations, measured discharge volume etc.) are unfortunately subject to distortion in both time and value axes. For this reason, it is necessary to find out suitable mechanism that is able to deal with this deformation. The Dynamic Time Warping (DTW) algorithm can be used for this purpose.

DTW is a technique to find an optimal alignment between two given sequences under certain restrictions (Muller 2007). The sequences are warped in a nonlinear fashion to match each other. First DTW was used for comparing two different speech patterns in automatic speech recognition. In information retrieval it has been successfully applied to dealing with time deformations and different speeds associated with time-dependent data. For our purposes, the DTW algorithm will be used as a tool for finding the longest common subsequence of two sequences.

4.1. Searching the mutual subsequences

Despite the fact that the DTW has its own modification for searching subsequences, it works perfectly only in a case of searching exact pattern in some signal database. This case is demonstrated in Figure 1, where sequence

s_1 = '3456' exactly matches the corresponding subsequence in sequence s_2 = '12**3456**789'. However, in real situations exact patterns are not available because they are surrounded by additional values (Figure 2a), or even repeated several times in the sequence (Figure 2b). Unfortunately, the basic DTW is not able to handle these situations and it fails or returns only a single occurrence of the pattern. To deal with this type of situations, own DTW modification was created. This modification is able to find the longest common time warped subsequences under selected restrictions. Because this paper's topic is not focused on modifications of the DTW, our proposed solution will be presented very briefly.

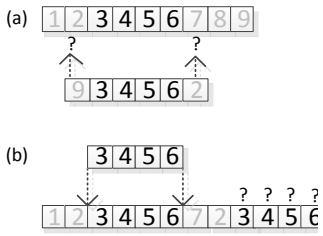

Figure 2: Basic DTW inaccuracies

The modification is based on the DTW's distance matrix (Muller 2007), in which we want to find the longest common warping paths with the lowest path cost as much as possible. The searching of warping paths starts by searching common column's and row's minimums. Formally, we are looking for points:

$$p(x,y): x = \min(c_x) \wedge y = \min(r_y), \quad (1)$$

where (x, y) are indices (coordinates) of minimum point, c_x is the x row and the r_y is the y column of distance matrix. Consider two sequences s_1 = *{4, 2, 3, 4, 5, 7}* and s_2 = *{5, 2, 3, 4, 8, 8}*, so their distance matrix will look like in Figure 3a.

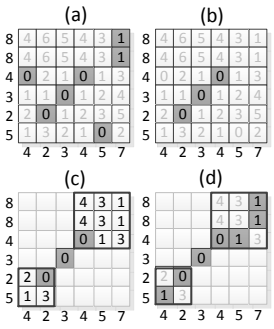

Figure 3: Distance matrices

Then, for each of these minimum cost points of the distance matrix:

1. If the actual point is contained in any other warping path, ignore this point and continue with the next one.
2. If there are some other local minimums in below, left or below left locations in the distance matrix, move there.
3. If the total cost of warping path is not overcome, add the current location to the warping path and continue with the point 2 until other minimums are available.
4. Now, no other close local minimums are available and the warping path for example looks like in the Figure 3b. However, if the maximum cost of warping is known, the warping path can be extended until the maximum path cost is reached. Extension of warping path means adding as much points as possible to the warping path from the first point of warping path towards the upper right corner of matrix, respectively from the last point of warping path towards the lower left corner of matrix.
5. For purpose of searching optimal extensions, classic DTW is used. Searching for extensions is actually searching of optimal warping paths in the new two submatrices showed in Figure 3c.
6. Once these paths are found, the original warping path is alternately extended in both directions until it reaches the maximum (see Figure 3d).

In paragraphs above, the cost of warping path was mentioned for several times. It is usual to specify the maximum cost of warping path as a constant, which cannot be overcome. However, tests showed that it is much suitable to specify the maximum cost of warping path as an average cost per warping point, because it is more immune to measurement errors and other distortions.

4.2. Post-Process Algorithm

The main idea of the Voting Experts DTW post-process is summarized into the following steps:

1. First of all, the high precision (but not complete) cuts are created by splitting the input with high level of threshold by the Two-Way Voting Experts method.
2. Let's suppose that there are m unique sequences which have been created according to the cuts from step 1.
3. A $m \times m$ distance matrix is build.
4. For each pair in this matrix, where the length of sequence s_1 is bigger than length of sequence s_2:
   (a) The optimal mapping of shorter sequence s_2 to the longer sequence s_1 is found by using DTW modified.
   (b) If the mapping cost does not overcome selected threshold, the longest sequence s_1 stores the shorter sequence s_2 into its own list of similar sequences. By this way,

every sequence gets its own list of the most similar shorter sequences.

(c) Each of the shorter sequences points to positions in the longer sequence, where it should be split. Because there is usually more than one similar shorter sequence, it is pointed to several locations whereas many of these locations are duplicated. For this reason, the votes are collected into internal vote storage.

(d) After these votes are collected, the local maximums are detected. These places are suggested as new cuts in original input.

5. The granted votes from step 4(d) are summed with votes of frequency and entropy experts in the input. Subsequently, the local maximums of votes are searched again. The cuts are made in locations where the number of granted votes is higher than the specified threshold.
6. Algorithm ends or it can continue with step 2 for further refinement.
7. After each post-process iteration, all found patterns can be received from the 4(b) step. Actually, the found patterns are groups of similar chunks.

For our algorithm improvement, several variants of each particular step were proposed and then their influences were tested on final results.

4.3. Post-Process Evaluation

Typical test of algorithm's verification of Voting Experts algorithm's performance is searching words in continuous text. In this text, spaces and punctuations (dots, dashes, new lines etc.) are removed and the goal of the algorithm is to put spaces back into correct places. Because the correct placement of spaces in the original text is known, it is very easy to quantify the algorithm's accuracy. For objective comparison we tested our proposed alternative approach in natural domain of Voting Experts – splitting continuous text (Kocyan, Martinovič, Kuchař, and Dvorský 2012). Results of the best algorithm configuration applied on various texts overcame almost all basic algorithm monitored quality indicators. Recall was improved by up to 18% and overal F-measure quality reached improvement about 5.5% in average. It is evident that our proposed alternative approach works with categorical data. A new challenge is to modify the algorithm for processing quantitative time series and to design suitable mechanism for Case-Base Reasoning's Retrieve step. And this is the main goal of this paper.

4.4. Test Collection

A very important aspect of every study is data collection. The data used in this paper consist of two parts: First of all, an artificial collection will be created and then it will be distorted. Using this collection, an algorithm will be explained, evaluated and the results will be presented. The second collection was obtained from the U.S. database - USGS (U.S. Geological Survey) Water Data for the Nation (section – surface data). Online access to this data is organized around the categories. We used discharge data (CFS - cubic feet per second) from stations located on the main rivers in the U.S.A. (years from 1986 to 2007, 30 min. step). This data was used for practical demonstration of proposed algorithm.

5. EXPERIMENTS

First of all, the algorithm was tested on artificial collection. This collection was generated from patterns displayed in Figure 4, which were randomly repeated to the total input's length of 200 elements.

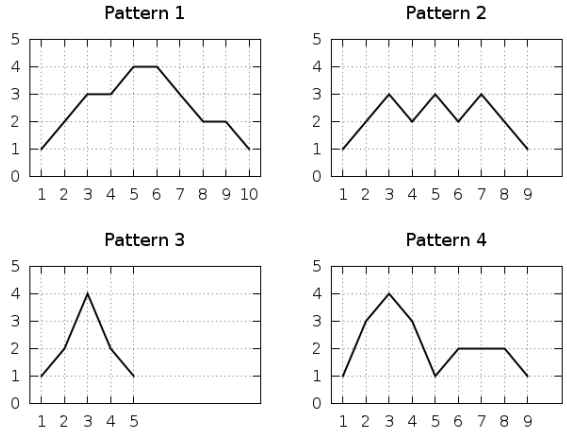

Figure 4: Set of Artificial Patterns

On this data collection, the original Voting Experts algorithm was executed with many configurations and the best result was taken. Then, with the same data, the proposed algorithm was tested in many configurations and also the best result was taken.

For the evaluation of proposed algorithm's performance, precision and recall coefficients were defined. Precision coefficient and recall coefficient rank among the most often used for the methods that are able to provide relevant documents in the information system. The precision coefficient is understood as the ratio of the amount of relevant documents returned to the entire number of returned documents. Recall represents the ratio of the amount of relevant documents returned to a given query to the entire amount of documents relevant to this query. In our case, the precision coefficient will be understood as the ratio of the amount of correct spaces induced by algorithm to the entire number of induced spaces. Recall will represent the ratio of the amount of correct induced spaces to the entire amount of spaces in input.

The basic algorithm reaches the best results with configuration listed in Table 1:

Table 1: The Best Configuration of Original VE

Parameter	Value
Sliding Window Size	10
Threshold for Cuts	4

To get high precision cuts for post-processing, the value of threshold had to be increased and followed by particular count of post-process cycles.

The best configuration of the post-process approach is shown in Table 2 and the comparison of both versions of algorithm's success is presented in Table 3.

Table 2: The Best Configuration of VE Post-Process

Parameter	Value
Sliding Window Size	10
Threshold for Cuts	9
Post-Process Cycles	13
Post-Process Threshold	8

Table 3: Evaluation of algorithms' success

Algorithm	Precision	Recall	F-Measure
Basic Voting Experts	0.71	0.68	0.70
VE with Post-Process	1	0.86	0.93

It is evident that proposed algorithm significantly outperforms the basic version of Voting Experts in all evaluation indicators and increases its performance.

Table 4 shows the progress of all indicators after each post-process cycle. The fact that the precision indicator still takes the value of 1 means that algorithm did not make any wrong cuts. Graphical representation of the table's values is in Figure 5.

Table 4: Progress of Evaluation Indicators

Post-Process Cycle	Precision	Recall	F-Measure
0 (VE output)	1	0.10	0.19
1	1	0.24	0.47
2	1	0.31	0.62
3	1	0.45	0.71
4	1	0.55	0.77
5	1	0.62	0.77
6	1	0.62	0.77
7	1	0.62	0.77
8	1	0.72	0.84
9	1	0.79	0.88
10	1	0.79	0.88
11	1	0.76	0.86
12	1	0.76	0.86
13	1	0.86	0.93

The second experiment was executed on distorted version of the previous data collection. Applied distortion randomly manipulated with length of patterns (elements of patterns were repeated or omitted) and its amplitude. Example of various distorted versions of one particular pattern can be seen in Figure 6.

The results of the experiment are shown in Table 5. It is evident that the original Voting Experts algorithm cannot work with distorted data and its accuracy rapidly decreases. Our proposed solution's accuracy decreases too, however it still provides better result with distorted input than the basic version on regular input. Examples of found patterns are shown in Figure 7.

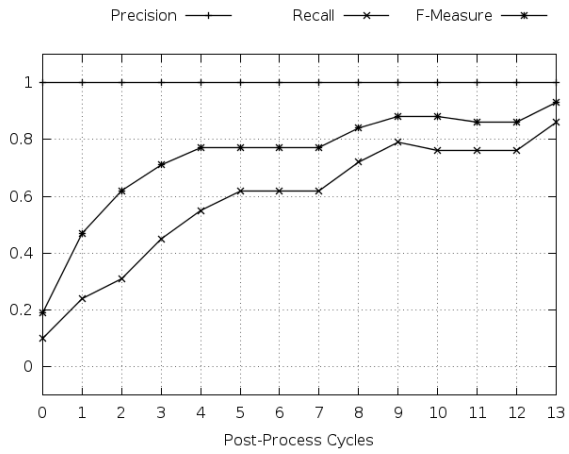

Figure 5: Progress of Evaluation Indicators

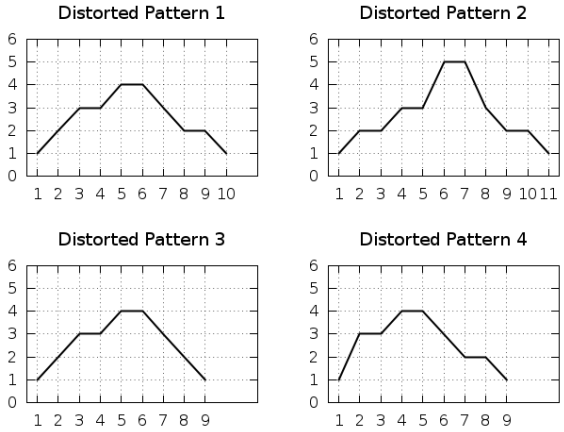

Figure 6: Example of Distorted Artificial Pattern

Table 5: Evaluation of algorithms using distorted input

Algorithm	Precision	Recall	F-Measure
Basic Voting Experts	0.48	0.58	0.53
VE with Post-Process	0.91	0.72	0.81

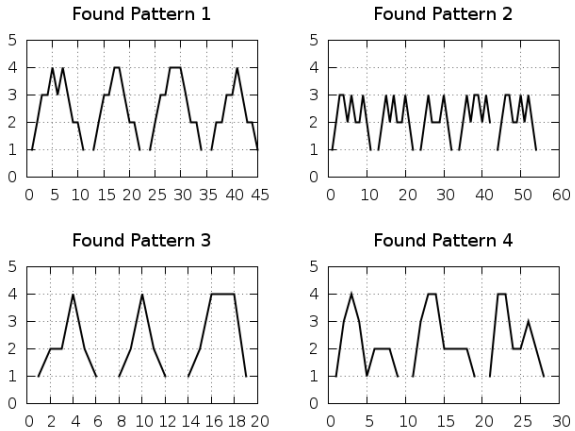

Figure 7: Example of Found Distorted Patterns

The last experiment was executed with river data described in paragraph 4.1. For this data collection, the exact real boundaries are not known, so it is impossible to numerically evaluate success of the algorithm. For this reason, the results of the algorithm will be

evaluated only visually and the examples of found patterns will be presented.

Before this data could be processed, it had to be converted to format suitable for the Voting Experts algorithm. For this reason, the data was encoded by SAX (Symbolic Aggregate approXimation) described in (Patel, Keogh, Lin, and Lonardi, 2002) and its dimension was reduced. Then, the algorithm was executed with the same configuration as in the example above. The found patterns are shown in Figure 8.

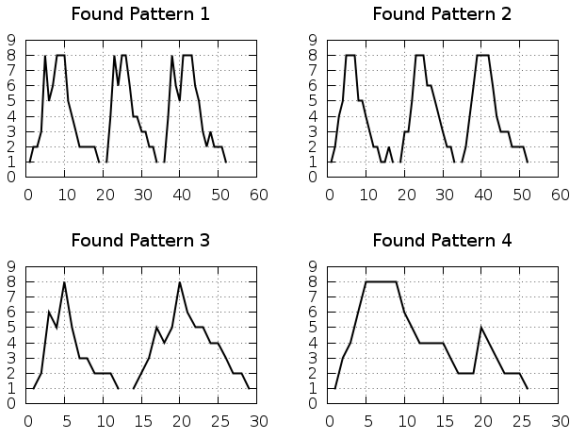

Figure 8: Example of Found River Patterns

6. CONCLUSION

The main goal of this paper was to provide suitable mechanism for searching characteristic patterns in order to improve the performance of Case-Based Reasoning. Found patterns can also be used for time series indexing or compressing.

Tests showed that proposed algorithm is able to successfully split input data into the meaningful episodes and then find characteristic patterns in both regular and distorted collections. Proposed alternative approach outperforms the original Voting Experts algorithm in all evaluation indicators (Precision, Recall, F-Measure) and moreover, it is more immune to working with distorted inputs.

Future research will be focused on optimizing and improving proposed algorithm's performance, especially on automatic settings of configuration's parameters, which have to be set manually for now. Additionally, we will focus on searching the universal encoding algorithms for transforming general time series into the format suitable for Voting Experts and on the modification of DTW for receiving characteristic patterns from a group of similar episodes.

ACKNOWLEDGMENTS

This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the grant project of the Technology Agency of the Czech Republic under number TA01021374, titled "Nove technologie ochrany zivotniho prostredi pred negativnimi nasledky pohybujicich se prírodnich hmot" (New Technologies for Environmental Protection against Negative Effects of Natural Mass Transfer).

REFERENCES

Cheng, J., Mitzenmacher, M., 2005. Markov Experts. *Proceedings of the Data Compression Conference (DCC)*.

Cohen, P. R., Adams, N., 2001. An Algorithm for Segmenting Categorical Time Series into Meaningful Episodes, *Proceedings of the Fourth Symposium on Intelligent Data Analysis, Lecture Notes in Computer Science*.

Cohen, P. R., Adams, N., and Heeringa, B, 2007. Voting Experts: An Unsupervised Algorithm for Segmenting Sequences. *In Journal of Intelligent Data Analysis*.

Comani, S., 1987. The historical temperature series of Bologna, Italy from 1716 to 1774. *Climatic Change 11(3)*: 375-390.

Fanta, B., Zaake, B.T., Kachroo, R.K., 2001. A study of variability of annual river flow of the southern African region. *Hydrological Sciences 46* (4): 513 – 524.

Giakoumakis, S. G., Baloutsos, G., 1997. Investigation of trend in hydrological time series of the Evinos River basin. *Hydrological Sciences Journal 42(1)*: 81-88.

Hewlett, D., Cohen P., 2009. Bootstrap Voting Experts. *Proceedings of the Twenty-first International Joint Conference on Artificial Intelligence (IJCAI).*

Kocyan, T., Martinovič, J., Kuchař, Š., and Dvorský J., 2012. Unsupervised Algorithm for Post-Processing of Roughly Segmented Categorical Time Series. *Proceedings of Databases, Texts, Specifications, and Objects*.

Maheras, P., Kolyva-Machera, F., 1990. Temporal and spatial characteristics of annual precipitation in the twentieth century. *Int. J. Climatol*. 10: 495-504.

Miller, M., Stoytchev, A, 2008. Hierarchical Voting Experts: An Unsupervised Algorithm for Hierarchical Sequence Segmentation. *Proceedings of the 7th IEEE International Conference on Development and Learning (ICDL)*.

Miller M., Wong P., and Stoytchev A., 2009. Unsupervised Segmentation of Audio Speech Using the Voting Experts Algorithm. *Proceedings of the Second Conference on Artificial General Intelligence (AGI)*.

Muller, M., 2007. Dynamic Time Warping. *Information Retrieval for Music and Motion*, Springer, ISBN 978-3-540-74047-6, 69--84.

Theodoridis, S., and Koutroumbas, K., 2006. Pattern Recognition. *Third Edition. Academic Press.*

Patel, P., Keogh, E., Lin, J., and Lonardi, S., 2002. Mining Motifs in Massive Time Series Databases. *In proceedings of the 2002 IEEE International Conference on Data Mining*. Maebashi City, Japan. Dec 9-12.

Watson, I., 1997. *Applying Case-Based Reasoning: Techniques for Enterprise Systems*, 15-38.