# Comparing Time Series through Event Clustering*

Juan A. Lara[1], Aurora Pérez[1], Juan P. Valente[1], and África López-Illescas[2]

[1] Facultad de Informática, Universidad Politécnica de Madrid,
Campus de Montegancedo, 28660, Boadilla del Monte, Madrid, Spain
j.lara.torralbo@upm.es, {aurora,jpvalente}@fi.upm.es
[2] Centro Nacional de Medicina del Deporte,
Consejo Superior de Deportes, C/ El Greco s/n, 28040, Madrid, Spain
africa.lopez@csd.mec.es

**Abstract.** The comparison of two time series and the extraction of subsequences that are common to the two is a complex data mining problem. Many existing techniques, like the Discrete Fourier Transform (DFT), offer solutions for comparing two whole time series. Often, however, the important thing is to analyse certain regions, known as events, rather than the whole times series. This applies to domains like the stock market, seismography or medicine. In this paper, we propose a method for comparing two time series by analysing the events present in the two. The proposed method is applied to time series generated by stabilometric and posturographic systems within a branch of medicine studying balance-related functions in human beings.

**Keywords:** Data Mining, Time Series, Event, Stabilometry, Posturography.

## 1 Introduction

Knowledge discovery in databases (KDD) is a non-trivial process that aims to extract useful, implicit and previously unknown knowledge from large volumes of data. Data mining is a discipline that forms part of the KDD process and is related to different fields of computing, like artificial intelligence, databases or software engineering. Data mining techniques can be applied to solve a wide range of problems, including time series analysis, which has come to be highly important in recent years.

A time series can be defined as a sequence X of time-ordered data $X = \{x_t, t = 1,\ldots,N\}$, where $t$ represents time, $N$ is the number of observations made during that time period and $x_t$ is the value measured at time instant $t$. Time series are usually represented as a graphs in a Cartesian system, where one of the axes represents time and the other (or others in the case of multidimensional series) records the value of the observation (Figure 1).

One interesting problem in the data mining field is the comparison of two time series. This calls for the determination of a measure of similarity indicating how alike two time series are. Most existing techniques compare one whole series with another whole series [1, 2]. However, there are many problems where it is requisite to focus on certain regions of interest, known as events, rather than analysing the whole time

* This work was funded by the Spanish Ministry of Education and Science as part of the 2004-2007 National R&D&I Plan through the *VIIP* Project (DEP2005-00232-C03).

J.M. Corchado et al. (Eds.): IWPACBB 2008, ASC 49, pp. 1–9, 2009.
springerlink.com 

series [3]. This applies, for example, to domains like seismography, where points of interest occur when the time series shows an earthquake, volcanic activity leading up to the earthquake or replications.

In this article, on the one hand, we propose a method that can locate similar events appearing in two different time series, that is, events that are similar and common to the two series, and, on the other hand, we also define a similarity measure between the two time series based on the idea that the more events they have in common the more alike they will be. This similarity measure will be needed to do time series clustering, pattern extraction and outlier detection.

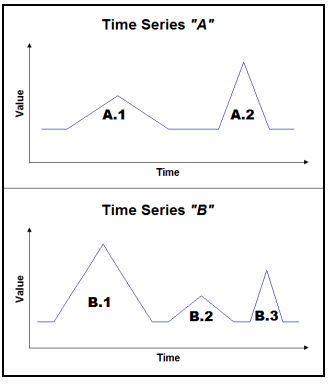


**Fig. 1.** Charts of two time series (*A* and *B*)

**Example.** Suppose that the events of a time series in a particular domain are the peaks generated by the local maxima. Given two time series, *A* and *B* (Figure 1), there are two regions of interest in series *A* and three in series *B*. Comparing the two series, we find that the first event in *A* (*A.1*) is very like the second in *B* (*B.2*), and the third event in *B* (*B.3*) is very like the second in *A* (*A.2*). In this case, the series *A* and *B* have two events in common and are, therefore, very alike.

The method developed throughout this article will be applied in the field of medicine known as stabilometry, which is responsible for examining balance-related functions in human beings. This is an important area within neurology, where the diagnosis and treatment of balance-related dysfunctions like *dizziness* has advanced enormously in recent years.

There is a device, called a posturograph, which is used to measure balance-related functions in human beings. The patient stands on a platform to do a number of tests (see Figure 2). We used a static *Balance Master* posturograph. In a static posturograph, the platform on which the patient stands, does not move. The platform has four sensors, one at each of the four corners: front-right (*FR*), front-left (*FL*), rear-right (*RR*) and rear-left (*RL*). While the patient is doing a test, each of the sensors receives a datum every 10 milliseconds. This datum is the intensity of the pressure that the patient is exerting on the above sensor. Therefore, at the end of the test, we have a time series composed of a set number of observations measured at different time points. Each of those observations can be viewed as a point in a four-dimensional space, as the value of each of the four above-mentioned sensors is measured at each time instant.

**Fig. 2.** Patient doing a test on the posturograph

The proposed method, which will be described in section 3, will be applied to this type of time series. The results of applying the method will be discussed in section 4, whereas the findings will be detailed in section 5. Before all this we will present work related to this article in section 2.

## 2 Related Work

There are many domains working with time series. Over the last few years, a lot of research has been carried out on the time series field in domains as far apart as medicine [4], the financial sector [5] or traffic control [6].

There are many techniques for comparing time series and extracting common subsequences. A technique for comparing times series based on the Fourier Transform is proposed in [1]. The aim is to extract a number of coefficients from the time series using the discrete Fourier transform, that is, by switching from the time to the frequency domain. Techniques based on alternatives to the Fourier transform, like the Wavelet transform [2, 7, 8], have been proposed.

The landmarks-based technique [9] proposes a method for comparing time series where the singular points (landmarks) of the curve, that is, the maximums, minimums and turning points, have to be stored. It proposes the use of six transformations that are applied to the landmarks. Several features of the landmarks are invariant to the application of certain transformations, and only the invariant features of each transformation are taken into account when looking for series that are similar under certain transformations.

Another type of approach employs the time warping technique. This is based on the idea that two series are similar if the distance between the two series when one of them is compressed on the time axis is less than a certain threshold [10, 11]. Another type of technique uses MBR (minimum bounding rectangles) to compare time series [12].

A method for discovering subsequences common to several series is proposed in [13]. It is based on the idea of building a tree storing all the possible common subsequences of length "*k*" at each depth level "*k*". The technique is very efficient thanks to the tree pruning process, which rejects solutions that do not meet certain minimum confidence conditions.

There are techniques, such as those proposed in [1] and [2], that are useful for comparing whole time series. In many domains, however, as is the case with stabilometry, it is more important to focus on certain events or regions of interest. There are different proposals in this respect, like Povinelli's system [3, 14]. Because of the increasing importance of events detection over the last few years, techniques and algorithms have been developed to detect events in complex time series [15].

In some domains, the value of not one but several observations might be measured at each time point, leading to multidimensional time series. This applies to the posturograph used in this research, as four values are recorded at each time instant. There are several techniques for studying multidimensional time series [16, 17].

In recent years, different and innovative proposals have emerged for data mining time series. Some of these methods use Markovian models to compare time series [18], others use graph theory [19], others again are based on comparing time series by looking at how they change shape [20], etc.

In this paper, we propose the application of computing techniques to medicine, something that has already been done in earlier work. Ever since the early expert systems, like Dendral [21] or Mycin [22], were first conceived, a host of medical decision support software systems have been developed [23, 24].

In [25] a system was developed capable of diagnosing injuries in top-competition athletes thanks to the analysis, using data mining techniques, of data generated by an isokinetics machine that measures the athletes' muscle strength when bending and stretching their members. Similarly, [4] proposes a technique to diagnose epilepsy, using data mining techniques.

The use of computing techniques in the domain of medicine has recently come to be common practice. However, the application of data mining techniques to posturographic data has a number of particularities that, taken together, single it out from their use in other domains. They are: (1) the structural complexity of the patient examinations, (2) the multi-dimensionality of the collected variables and (3) the fact that relevant information appears in definite regions of each series and not across the whole series.

## 3 Proposed Method

The method presented here is applicable for domains where the points at which a particular event takes place and not the whole time series are of interest. An event is part of the time series that starts at one point and ends at a later point and is of interest for the expert in the domain in question. Many state-of-the-art techniques, like the Discrete Fourier Transform of the Wavelet Transform, compare two time series as a whole. However, these techniques are not suited for those domains where only small parts of the time series are relevant. These events can occur at any instant and with random duration. The method described in this paper focuses on this kind of time series.

Note that the reference domain in this article is stabilometry. The study run focused on one of the tests run on the posturograph: the *UNI* test. This 10-second test aims to measure how well the patient is able to keep his or her balance when standing on one leg with either both eyes open or both eyes shut. While the patient is doing the test, each sensor sends a datum to the central computer every 10 milliseconds. This datum is the intensity or pressure that the patient is putting on that sensor. Therefore, at the end of the test, we have a time series storing four values for each time instant (see Figure 3).

An ideal test would be one where the patient kept a steady stance and did not wobble at all throughout the whole test. The interesting events of this test occur when the patient loses balance and puts the lifted leg down onto the platform. This type of event is known in the domain as a *fall*. When there is a fall, the respective sensors for the lifted leg will register the pressure increase. Figure 3 shows the time series of a patient who has done the UNI test. The curves at the top of the figure are the values recorded by the RR and RF sensors, that is, the right-leg sensors, the leg the patient was standing on. The curves at the bottom of the figure are the values recorded by sensors LR and LF, that is, the left-leg sensors, the leg that should be lifted. The pressures peaks generated when there is a fall event are highlighted.

The next step after identifying the fall events is to determine a similarity measure between two multidimensional time series by obtaining and comparing the fall events that appear in both time series.

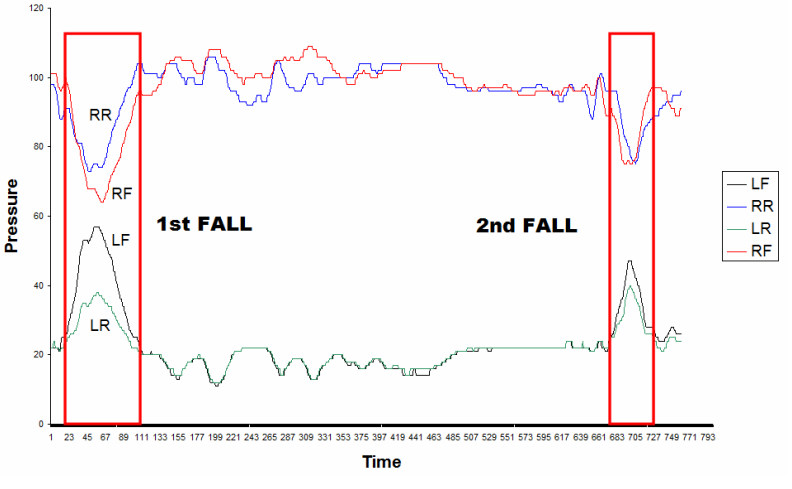


**Fig. 3.** UNI test time series, highlighting two events (*falls*)

Formally, the aim is to find a function $F$ that takes two times series $A$ and $B$ and returns a similarity value in the interval [0,1], where 0 indicates that the two series are completely different and 1 denotes that the two series are identical, as described in equation (1).

$$F : ST, ST \rightarrow [0,1]$$
$$F : A, B \rightarrow Similarity(A, B) \tag{1}$$

To determine similarity, the proposed method looks for events that appear in both series. The greater the number of events that the two series to be compared have in common, the closer similarity will be to 1. If the series do not have any event in common, similarity will be equal to 0.

To determine whether an event in one time series appears in another, the event has to be characterized by means of a set of attributes and compared with the other events of the other series. To speed up this process, all the events present in the two time series are clustered. Therefore, if two events belong to the same cluster, they are similar. The goal is to find events that are members of the same cluster and belong to different time series.

Therefore, the proposed algorithm for extracting events common to two time series *A* and *B* is:

1. **Extract all the events $E_j$ of both series (events that appear in *A* or in *B*) and characterize each event by means of a set of attributes.** This point is domain dependent, as event characterization will depend on the type of time series. For example, the events of interest in the reference domain we are using are *falls* and they are characterized by the following attributes:
   a) Region in which the lifted leg falls.
   b) Intensity of the pressure exerted by the falling patients' foot on the platform and drop in the intensity of pressure of the standing leg sensors.
   c) Time from when the patient starts to lose balance until he or she falls.
   d) Time from when the patient falls to when he or she recovers.
2. **Cluster all the events extracted in point 1.** To do this, it is necessary to calculate the distance between each pair of events explained under step 1 of the algorithm. We opted to use the city-block distance. This distance calculates the sum of the absolute differences of each of the coordinates of two vectors:

$$d_{ij} = \sum_{k=1}^{p} \left| x_{ik} - x_{jk} \right| \tag{2}$$

   In equation (2), *i* and *j* are the vectors to be compared and *p* is the number of coordinates (dimension). We looked at other distance measures, but finally opted to use the city-block distance, because the mean distance per attribute is used during the clustering process to determine whether two elements are members of the same cluster. This mean distance per attribute is obtained straightforwardly by dividing the total distance $d_{ij}$ by the number of attributes *p*.
3. **For each cluster from step 2 and as long as there are events from the two series in the cluster:**
   3.1. **Create all the possible event pairs $(E_i, E_k)$ for which $E_i \in A$ and $E_k \in B$.**
   3.2. **Select the event pair that minimizes *distance* $(E_i, E_k)$.** Equation (2) describes the distance to be used.

*(This extracts the two events that are in the same cluster, belong to different time series and are the most alike)*

3.3. **Delete events $E_i$ and $E_k$ from the cluster.**

3.4. **Return the pair $(E_i,E_k)$ as an event common to both series.**

By the end of this process, we will have managed to extract the event pairs that are similar in the two series. This is a key point for the mechanism that establishes how alike the two time series being compared are.

A common event, $C_i$ is a pair $C_i = (E_i,E_k) \mid E_i \in A$, $E_k \in B$, output by step 3.4 of the algorithm. If $E = \{E_j, j=1..n\}$ is the set of all events present in $A$ or in $B$ (output by step 1 of the algorithm) and $C=\{C_i, i=1..m\}$ is the set of common events present in both series, $A$ and $B$, then the similarity can be obtained by comparing the amount of the time series that is common to the two time series ($C_i$) with the total amount of the time series of interest ($E$). The more events the series to be compared have in common, the closer the similarity will be to 1.

## 4 Results

The system implementing the described method has been evaluated by running a battery of tests. These tests were done on time series generated by a posturographic device. The study focused on the data from the *UNI* test. For this test, the events of interest are *falls* that take place at times when the patient suffers a severe loss of balance leading him or her to put down the leg that he or she should be lifting.

We received support for the evaluation from the *High Council for Sports*, an institution attached to the *Ministry of Education and Science*, responsible for coordinating sporting activities in Spain. This institution provided times series for 10 top-competition athletes of different sexes, ages and sports for this study. Note that each time series is composed of four dimensions. At this early stage of the project, 10 is a reasonable number of patients, taking into account how difficult is to obtain this information due to the shortage of top-competition athletes, the complexity of the tests and the fact that there is no public posturographic database. An expert from the above institution helped to validate the results generated by implementing the proposed method. We had to rely on only one expert because stabilometry is a very new field and there are not many experts in this area. In actual fact, there are only a couple of stabilometric devices in use in Spain.

The evaluation of the research focused on one point: *Are the comparisons made by the system of similar quality to those made by the expert?*

To evaluate this point, all time series were compared with each other (if there are 10 time series, and each time series is compared with all the others except itself, we have a total of 45 comparisons). For each of the above comparisons, the similarity rating generated by the method was checked against the similarity score determined by the expert. In each comparison, the expert was asked to determine a similarity rating from the following: {*Not at all similar, Not very similar, Moderately similar, Fairly similar, Very similar*}.

The rating *Not at all similar* would correspond to a similarity in between the interval [0, 0.2), the rating *Not very similar* would correspond to the interval [0.2, 0.4), and

so on up to the rating *Very similar*, which would correspond to a similarity score in [0.8, 1].

When evaluating a comparison, the agreement between the expert and the method could be *Total* if the similarity interval is the same in both cases, *Very High* if the interval determined by the system and by the expert are adjacent, and *Low* in any other case.

The results of the comparisons by the expert and the system were also good, as, agreement between the system and the expert was *Total* or *Very High* in 39 out of 45 cases. Only 6 of the cases showed some differences between the results generated by the system and the ratings determined by the expert.

## 5 Conclusions

We have developed a method to compare time series by matching up their relevant events. This method is suitable for domains where the relevant information is focused on specific regions of the series, called events, and where the remaining regions are not relevant.

The method was evaluated on time series for top-competition athletes. After performing the different evaluation tests, the results were considered very satisfactory for both the research team and the expert physicians, boosting their will to develop further cooperation in this field.

This project is at a very early stage. The method we have developed is a preliminary version. In the future we intend to refine the method using other distance measures, and apply this method to time series from some other domains.

## References

1. Agrawal, R., Faloutsos, C., Swami, A.: Efficient Similarity Search In Sequence Databases. In: FODO. Evanston, Illinois (1993)
2. Chan, K., Fu, A.W.: Efficient Time Series Matching by Wavelets. In: ICDE, pp. 126–133. Sydney-AUS (1999)
3. Povinelli, R.: Time Series Data Mining: identifying temporal patterns for characterization and prediction of time series, PhD. Thesis. Milwaukee (1999)
4. Chaovalitwongse, W.A., Fan, Y., Sachdeo, R.C.: On the Time Series K-Nearest Neighbor Classification of Abnormal Brain Activity. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 1 (2007)
5. Lee, C.L., Liu, A., Chen, W.: Pattern Discovery of Fuzzy Time Series for Financial Prediction. IEEE Transactions on Knowledge and Data Engineering 18(5) (2006)
6. Yin, J., Zhou, D., Xie, Q.: A Clustering Algorithm for Time Series Data. In: Proceedings of the Seventh International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT 2006). IEEE, Los Alamitos (2006)
7. Tseng, V.S., Chen, C., Chen, C., Hong, T.: Segmentation of Time Series by the Clustering and Genetic Algorithms. In: Sixth IEEE International Conference on Data Mining - Workshops ICDMW (2006)

8. Kumar, R.P., Nagabhushan, P., Chouakria-Douzal, A.: WaveSim and Adaptive WaveSim Transform for Subsequence Time-Series Clustering. In: 9th International Conference on Information Technology, ICIT (2006)
9. Perng, C., Wang, H., Zhang, S.R., Parker, D.S.: Landmarks: A New Model for Similarity-Based Pattern Querying in Time Series Databases. In: ACDE, San Diego, USA, pp. 33–44 (2000)
10. Rafiei, D., Mendelzon, A.: Similarity-Based Queries for Time Series Data. In: ACM SIGMOD, Tucson, AZ, pp. 13–25 (1997)
11. Park, S., Chu, W., Yoon, J., Hsu, C.: Efficient Searches for Similar Subsequences of Different Lengths in Sequence Databases. In: ICDE, San Diego, USA, pp. 23–32 (2000)
12. Lee, S., Chun, S., Kim, D., Lee, J., Chung, C.: Similarity Search for Multidimensional Data Sequences. In: ICDE, San Diego, USA, pp. 599–610 (2000)
13. Alonso, F., Martínez, L., Pérez, A., Santamaría, A., Caraça-Valente, J.P.: Integrating Expert Knowledge and Data Mining for Medical Diagnosis. In: Expert Systems Research Trends, cap. 3, pp. 113–137. Nova Science Ed. (2007)
14. Povinelli, R., Feng, X.: A New Temporal Pattern Identification Method for Characterization and Prediction of Complex Time Series Events. IEEE Transactions on Knowledge and Data Engineering 15(2) (2003)
15. Vilalta, R., Sheng, M.: Predicting rare events in temporal domain. In: IEEE International Conference on Data Mining, pp. 474– 481 (2002)
16. Faloutsos, C., Ranganathan, M., Manolopoulos, Y.: Fast Subsequence Matching in Time-Series Databases, pp. 4190–429. ACM SIGMOD (1994)
17. Kahveci, T., Singh, A., Gürel, A.: Shift and scale invariant search of multi-attribute time sequences, Technical report, UCSB (2001)
18. Wang, Y., Zhou, L., Feng, J., Wang, J., Liu, Z.: Mining Complex Time-Series Data by Learning Markovian Models. In: Proceedings of the Sixth International Conference on Data Mining ICDM 2006. IEEE, Los Alamitos (2006)
19. Jan, L., Vasileios, L., Qiang, M., Lakaemper, W.R., Ratanamahatana, C.A., Keogh, E.: Partial Elastic Matching of Time Series. In: Proceedings of the Fifth IEEE International Conference on Data Mining (2005)
20. Dong, X., Gu, C., Wang, Z.: Research On Shape-Based Time Series Similarity Measure. In: Proceedings of the IEEE Fifth International Conference on Machine Learning and Cybernetics. Dalian (2006)
21. Lederberg, J.: How Dendral Was Conceived and Born. In: ACM Symposium on the History of Medical Informatics, November 05 1987. National Library of Medicine, Rockefeller University (1987)
22. Shortliffe, E.H.: Computer Based Medical Consultations: MYCIN. American Elsevier, Amsterdam (1976)
23. Edberg, S.C.: Global infectious diseases and epidemiology network (GIDEON): A world wide web-based program for diagnosis and informatics in infectious diseases, Clinical Infectious Diseases. official Publication of the Infectious Diseases Society of America 40(1), 123–126 (2005)
24. Gil, D., Soriano, A., Ruiz, D., Montejo, C.A.: Embedded systems for diagnosing dysfunctions in the lower urinary tract. In: Proceedings of the 22nd Annual ACM Symposium on Applied Computing (2007)
25. Alonso, F., Caraça-Valente, J.P., González, A.L., Montes, C.: Combining expert knowledge and data mining in a medical domain. Expert Systems with Applications 23, 367–375 (2002)