Dipartimento di Informatica
Università del Piemonte Orientale "A. Avogadro"
Spalto Marengo 33, 15100 Alessandria
http://www.di.unipmn.it

TECHNICAL REPORT
TR-INF-2004-03-05-UNIPMN
(March 2004)

RHENE: A Case Retrieval System for Hemodialysis Cases with Dynamically Monitored Parameters

Author: S. Montani, L. Portinale, R. Bellazzi, G. Leonardi
({stefania,portinal}@unipmn.it)

RHENE: A Case Retrieval System for Hemodialysis Cases with Dynamically Monitored Parameters

Stefania Montani[1], Luigi Portinale[1], Riccardo Bellazzi[2], Giorgio Leonardi[2]
[1] Dipartimento di Informatica, Università del Piemonte Orientale, Italy
[2] Dipartimento di Informatica e Sistemistica, Università di Pavia, Italy

March 22, 2004

Abstract

In this paper, we present a case-based retrieval system called RHENE (**R**etrieval of **HE**modialysis in **NE**phrological disorders) working in the domain of patients affected by nephropatologies and treated with hemodialysis. Defining a dialysis session as a case, retrieval of past similar cases has to operate both on static and on dynamic features, since most of the monitoring variables of a dialysis session are time series. In RHENE, retrieval relies upon a multi-step procedure. In particular, a preliminary classification step, based on static features, reduces the retrieval search space. Intra-class retrieval then takes place by considering dynamic features, and is articulated as follows: (1) "locally" similar cases (considering one feature at a time) are extracted and the intersection of the retrieved sets is computed; (2) "global" similarity is computed - as a weighted average of local distances - and the best cases are listed. The main goal of the paper is to present an approach for efficiently implementing step (2), by taking into account specific information regarding the final application. We concentrate on a classical dimensionality reduction technique for time series allowing for efficient indexing, namely Discrete Fourier Transform (DFT). Thanks to specific index structures (i.e. *k-d trees*) range queries (on local feature similarity) can be efficiently performed on our case base; as mentioned above results of such local queries are then suitably combined, allowing the physician to exhamine the most similar stored dialysis sessions with respect to the current one. The system can be seen as a support for patient examination and therapy evaluation, but could also be adopted as a means for assessing the quality of the overall hemodialysis service, providing a useful input from the knowledge management perspective.

1 Introduction

Health Care Organizations (HCO) have nowadays evolved into complex enterprises, in which the management of knowledge and information resources is a key success factor in order to improve their efficacy and efficiency. Unfortunately, although HCO are data-rich organizations, their capability of managing implicit (i.e. operative) knowledge is still very poor: the day-by-day collection of patients' clinical data, of health care provider actions (e.g. exams, drug deliveries, surgeries) and of health care processes data (admissions, discharge, exams request) is not often followed by a thorough analysis of such kind of information. Thanks to the Knowledge Management (KM) perspective [5], on the other hand, it is now clear that implicit knowledge may be effectively used to change organizational settings and to maintain and retrieve unstructured situation-action information [12]. In recent years, Case-Based Reasoning (CBR) has become widely accepted as a useful computational instrument for KM; the retrieval and reuse of past data and the possibility of retaining new information fit the KM objectives of keeping, increasing and reusing knowledge with particular attention to decision making support [5].

In medical applications, in particular, analogical reasoning is typically applied for decision making: physicians use to reason by recalling past situations, afforded by themselves or by some colleague and this kind of process is often biased by the tendency of recalling only the most recent cases. The CBR methodology could be of great help, since it enables an automatic retrieval of *all relevant* past situation-action patterns (including the oldest ones), as well as the retrieval of the other physicians' expertise, embedded into concrete examples [11].

Despite CBR appears to be an appropriate technique to support medical decisions, its exploitation in this field has not been as successful as in other domains; probably the weakness resides in the difficulties of implementing the adaptation step of the CBR cycle [1], being adaptation strongly application-dependent. As a matter of fact, the definition of a (possibly general) framework for performing adaptation in medical problems is a challenging task. Moreover, physicians would not easily accept a therapy/diagnosis automatically produced by a decision support system. On the other hand, a pure retrieval system, able to extract relevant knowledge, but that leaves the user the responsibility of providing an interpretation of the current case and of proposing a solution, seems much more suitable in this context.

The goal of this paper is to present RHENE[1] (**R**etrieval of **HE**modialysis in **NE**phrological disorders) a case-based system, developed in order to investigate the application of retrieval techniques in a time-dependent clinical domain: the management of End Stage Renal Disease (ESRD) patients treated with hemodialysis. Even though the system concentrates only on case retrieval, its architecture is non-trivial.

In particular, a multi-step procedure is implemented, where retrieval itself is anticipated by a *classification* phase. Classification provides a reduction of the retrieval search space, by identifying relevant subparts of the case base. In particular, the procedure can be automatic (the system implements a k-Nearest Neighbour (k-NN) approach on a subset of the case features), or user driven (the physician explicitly selects on which subparts of the library s/he wants to concentrate the attention).

Intra-class *retrieval* is then performed. Retrieval is in turn structured in a multi-step

[1]RENE in Italian means *kidney*.

fashion: *local similarity* is first taken into account, by allowing the selection of a subset of very relevant features, on which a range query (one in each feature's direction) is executed. The intersection of the locally similar cases is then computed, thus extracting the cases that satisfy the request of being within all the specified ranges of similarity contemporaneously. Clearly this is a strong requirement, but results can be finely tuned by varying the range parameters. *Global similarity* is then computed, as a weighted average of local similarities in the space of all the case features (including classification ones). In this way, the best cases are identified and ranked.

In particular, since in the hemodialysis domain most of the case features are in the form of time series, the (local) retrieval step requires a pre-processing phase, in which dimensionality reduction techniques are resorted to, in order to speed up the retrieval process itself, while maintaining sufficient information about the series and avoiding false dismissals.

Finally, retrieval takes advantage of an index structure, built on the series coefficients, that allows to avoid exhaustive search. A range query algorithm directly operating on the index has been implemented [13].

The paper is organized as follows: section 2 provides some details about the application domain, while section 3 addresses the technical aspects of our approach, by describing the basic RHENE architecture with some examples of useful retrieval of dialysis sessions; conclusions are discussed in section 4.

2 Hemodialysis treatment for ESRD

ESRD is a severe chronic condition that corresponds to the final stage of kidney failure. Without medical intervention, ESRD leads to death. Hemodialysis is the most widely used treatment method for ESRD; it relies on an electromechanical device, called hemodialyzer, which, thanks to an extracorporal blood circuit, is able to clear the patient's blood from metabolites, to re-establish acid-base equilibrium and to remove water in excess. On average, hemodialysis patients are treated for four hours three times a week. Each single treatment is called a hemodialysis (or simply a dialysis) session. Hemodialyzers typically allow to collect several variables during a session, most of which are in the form of time series (see Table 1); a few are recorded in the form of single data points (see Table 2 for some of them). As regards time series, in the current technical settings the sampling time ranges from 1 min to 15 min.

The most important analysis is to evaluate the agreement of the dialysis session to the prescribed therapy plan; in fact, sessions are classified as: **type 1**, positive session that agree to the therapy plan without external (from hospital attendants) intervention; **type 2**, positive session after attendants' intervention; **type 3**, negative session that fail to adhere to the therapy plan. In this context, the application of case-based retrieval techniques seems particularly suitable for hemodialysis efficiency assessment. In particular, defining a dialysis session as a case, it is possible to retrieve cases with the same outcome, or, more in detail, to look for similar situations - typically patterns correspond-

Variable name	Abbreviation
Venous Pressure	VP
Blood Bulk Flow	QB
Arterial Pressure	AP
Systolic Pressure	SP
Diastolic Pressure	DP
Cardiac Frequency	CF
Hemoglobin	Hb
Hematic Volume	HV
Output Pressure of dialyzer	OP
Dialysate Conductivity	DC

Table 1: Monitoring variables for hemodialysis, collected as time series.

Variable name	Abbreviation
Weight Before Session	WB
Weight Loss	WL
Dry Weight	DW
Vascular Access	VA
Dialysis Time duration	T

Table 2: Monitoring variables for hemodialysis, collected as single data points (one per session).

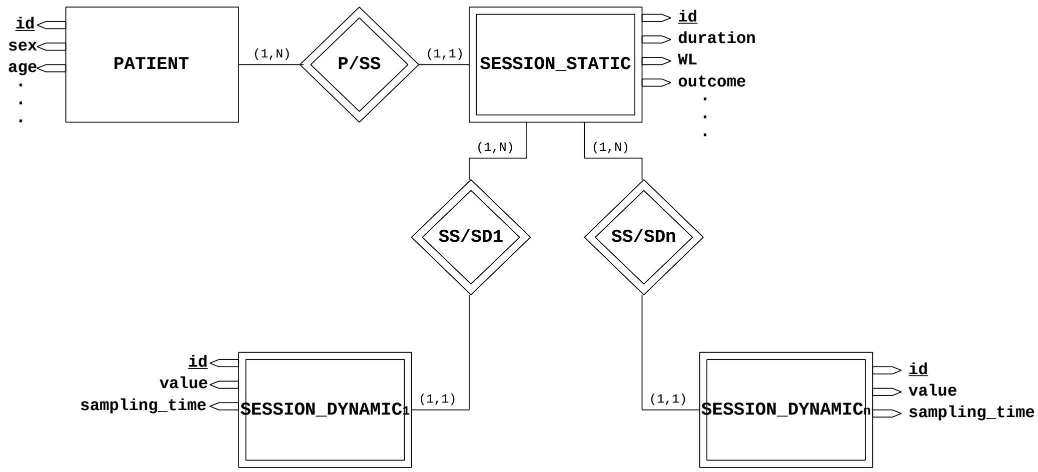

Figure 1: Case structuring in RHENE

ing to persistent failures over time. It is then possible to highlight if these patterns are repeated over the same patient or over different ones and what solutions have been provided in those cases, in terms of dialysis prescription (i.e. the prescribed flow rates at the beginning of dialysis).

3 RHENE: a Case Retrieval Stystem for ESRD

3.1 Basic Architecture

As previously observed, in our application, a dialysis session is interpreted as a case. The case structure involves two categories of features:

- *static features*, representing: (i) general information about the patient such as age class, sex, type of the disease that caused ESRD; (ii) long-term varying data about the patient, that can be approximately considered as static within an interval of a few weeks/months (e.g. several laboratory exams); finally (iii) general information about the dialysis session and the dialysis targets such as the dry weight - i.e. the desired weight at the end of a dialysis session, the vascular access, the dialysis duration and the additional pharmacological treatments (see Table 2);

- *dynamic features* which are the information automatically recorded within a dialysis session in the form of time series with different sampling times (see Table 1).

Moreover, each case records the *outcome* of the session, following the classification outlined above (type $1, 2$ and 3).

Case structure is described by the Entity-Relationship diagram of figure 1; time invariant information about each patient and the whole dialysis session are mantained in the PATIENT and SESSION_STATIC entities respectively, while time series information are maintained in the SESSION_DYNAMIC$_i$ entities corresponding to session dynamic feetures. Note that the case outcome is a static information and it is therefore stored in the SESSION_STATIC entity.

Since static features provide both a general characterization of the patient and of the dialysis session as a whole, they provide the context under which to evaluate the results of the dialysis, based on the interpretation of all the relevant measured parameters (both static and dynamic). Following some classical CBR literature [7], defining the context for retrieval corresponds to the so-called *situation assessment* step of CBR. Therefore, it is quite natural to structure case retrieval as a two-step procedure, articulated as follows:

1. **Classification**. Produces the relevant context under which to base retrieval; a classification step can be important if the physician needs to restrict attention only to particular subsets of the whole case base.

2. **Retrieval**. Takes place on the restricted case base possibly produced by the classification step (intra-class retrieval); in our system, it is in turn a two-step procedure: first a local retrieval in the space of each single dynamic feature is performed, by exploiting a range query on the corresponding index, finally local retrievals are suitably combined, in order to produce the set of most similar cases to the current one.

In the following, we will motivate and detail our approach, with particular emphasis on intra-class retrieval.

3.2 Classification

The classification step is implemented relying on static features alone. Target classes can be both implicit or explicit. In the first situation, there is no need to explicitly identify a set of predefined classes, but a k-NN step is used, in order to restrict the case library. This makes sense for example when a classification is required at the PATIENT entity level; static features of the patient are used in order to obtain (through *k-NN retrieval*) the set of most similar patient: only cases related to such patients are then used in intra-class retrieval.

This approach can be in principle adopted also when considering both patient and session static features (or session static features alone); however, in this case it is more reasonable to exploit a set of predefined explicit classes[2]. The explicit target classes can be identified in the different diseases patients are affected with or in particular characterizations of patients concerning age, sex, weight, life style or in characterization of sessions concering duration, drug treatments, hemodialysis treatment modality, etc... Cases belonging to the same class as the input case are identified this time through *k-NN classification* and they will be used as the search space for the subsequent retrieval step.

[2]Indeed, if we perform k-NN retrieval on session static feature, too few cases will be kept as search space, unless to consider large values for k.

Finally, the classification step can also be realized manually by the physician if s/he wants to directly restrict her/his attention to particular cases; for instance s/he can be interested only in cases belonging to the same patient in the latest month, or only in cases of patients following a given diet, etc...

In order to implement this step, in RHENE we resort to the standard Heterogeneous Euclidean-Overlap Metric (HEOM) [14], with the use of distance tables in case of nominal features.

3.3 Intra-class retrieval

Intra-class retrieval is the core of our methodology. At the beginning of the consultation, the physician has the possibility of choosing a set of dynamic features (that are in the form of time series) on which to ground the retrieval; this allows her/him to focus the attention on a subset of features that s/he considers relevant for the analysis s/he's going to perform. The requirement implemented by the system is that the retrieved cases must have a required level of similarity for every selected feature. For each one of the selected dynamic features (a subset of those listed in Table 1), we work on *local similarity*, i.e. we look for the most similar cases to the input case relatively to the direction represented by the feature at hand. Local results are then combined and a set of complete cases is returned, ranked by global similarity with respect to the target one (see below).

A wide literature exists about similarity-based retrieval of time series. Several different approaches have been proposed (see the survey in [6]), but most are based on the common premise of dimensionality reduction. The reduction of the time series dimensionality should adopt a transform that preserves the distance between two time series or understimates it. In the latter case a post-processing step is required to filter out the so-called "false alarms"; the requirement is never to overstimate the distance, so that no "false dismissals" can exist [6]. A widely used transform is the Discrete Fourier Transform (DFT) [2].

DFT maps time series to the frequency domain. DFT application for dimensionality reduction stems from the observation that, for the majority of real-world time series, the first (1-3) Fourier coefficients carry the most meaningful information, and the remaining ones can be safely discarded. Moreover, Parseval's theorem [10] guarantees that the distance in the frequency domain is the same as in the time domain, when resorting to any similarity measure that can be expressed as the Euclidean distance between feature vectors in the feature space. In particular, resorting only to the first Fourier coefficients can understimate the real distance, but never overestimates it.

In our system, we are currently implementing DFT as a means for dimensionality reduction, exploiting the Euclidean distance as a similarity measure (in particular, in presence of missing data, we set the distance equal to its maximum value, i.e. to the feature range). The choice of DFT is motivated by the observation that DFT is a standard technique. Moreover, DFT offers the possibility of relying on well known index structures, without studying ad hoc solutions and avoiding exhaustive search. In particular, we have implemented an index belonging to the family of *k-d trees* and a range query algorithm directly operating on k-d trees themselves [13].

Note that, if every distance is in the range $[0,1]$ independently of the considered feature f, it becomes more natural to characterize a range query, and we can exploit a set of parameters $0 \leq s_i \leq 1$ as the distance thresholds for the various range queries concerning the dynamic features of our cases.

In order to make the distance scale independent with respect to the series values, we operate as follows. Given two time series $X = \{x_1, \ldots x_r\}$ and $Y = \{y_1, \ldots y_q\}$, a parametric distance measure can be defined by considering an integer parameter p (if $p = 2$ we get the standard Euclidean distance):

$$D(X, Y, p) = \left(\sum_{j=1}^{\min(r,q)} |x_j - y_j|^p \right)^{\frac{1}{p}}$$

Distance can then be normalized over the range $RANGE_f$ of the corresponding feature f:

$$D_f(X, Y, p) = \left(\frac{1}{m} \sum_{j=1}^{m} \left| \frac{x_j - y_j}{RANGE_f} \right|^p \right)^{\frac{1}{p}} = \frac{D(X, Y, p)}{m^{\frac{1}{p}}\ RANGE_f}$$

with $m = \min(r, q)$.

In details, given a query case C_Q, intra-class retrieval starts by considering each single dynamic feature f that the physician has selected for her/his analysis; let T_f be the k-d tree index for feature f, Q_f the query series (i.e. the time series relative to feature f in case C_Q) and s the distance threshold for the range query (obviously, for a non-selected feature it is sufficient to apply the same mechanism and to perform the range query with $s = 1$). The following steps are then implemented: since the dialysis device has starting and ending phases during which monitored data are meaningless, the query series Q_f is first validated by removing head and tail data corresponding to noisy values; in this way all the considered time series are aligned to the first valid point. After that, Q_f is reduced through DFT by considering a predefined number of coefficients[3] (usually from 3 to 6). We are then able to perform a range query on T_f using Q_f and the threshold s; this returns a set of time series (relative to feature f) having a distance from Q_f that may be less than s (due to Parseval's theorem we are only guaranteed that no indexed time series whose distance is actually less than s has been missed). We then need a post-processing of the results, where actual distance with respect to Q_f is computed.

The whole process is performed for every dynamic feature that has been selected for local retrieval and finally only cases that have been retrieved in every feature direction are returned (case intersection). Figure 2 depicts this mechanism. As a matter of fact, the case intersection step first extracts from the case library the whole case to which the series belongs and then perform the intersection of the obtained set of cases. In this way we are guaranteed that returned cases have a distance less than the threshold for every considered dynamic feature (as mentioned above).

[3]The number of DFT coefficients to consider is a tunable parameter of the system.

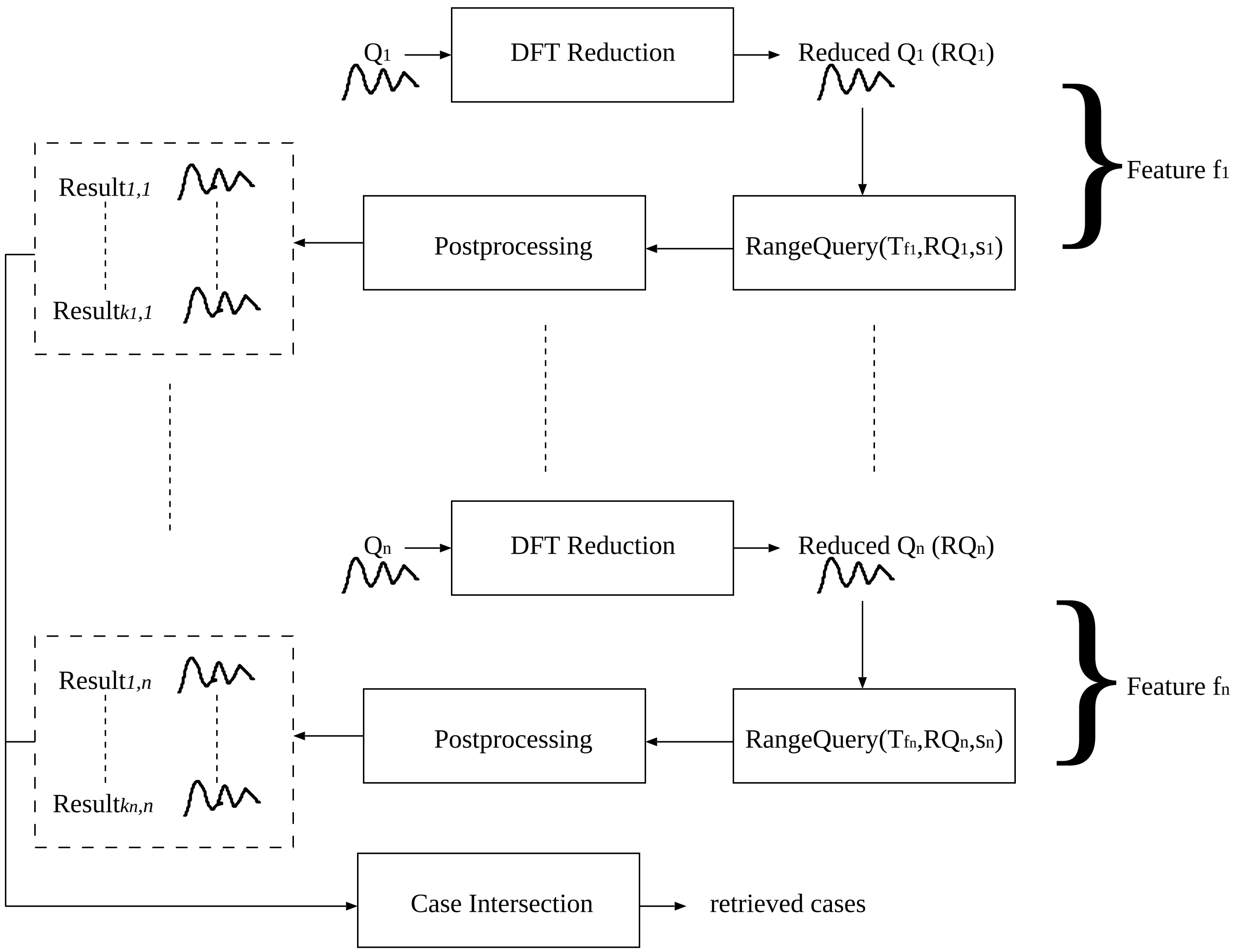

Figure 2: Block scheme for local retrieval.

Since we are finally interested in obtaining the best cases in terms of global distance with respect to C_Q, we compute such a global measure as a weighted average of feature (local) distances for every returned case:

$$D(C, C_Q) = \frac{\sum_{i=1}^{n} w_i D_{f_i}(X_C, Y_Q, p)}{\sum_{i=1}^{n} w_i}$$

where C is a retrieved case, C_Q the query case, X_C and Y_Q are the time series (values) of the feature f_i in case C and C_Q respectively and w_i is the weight representing the importance of feature f_i; the latter is another tunable parameter of the system available to the physician for biasing the order of presentation with more emphasis on a particular set of features (usually those selected for local retrieval, since they represent the features on which to base the analysis of the results).

3.4 Some retrieval examples

We tested the retrieval system on a set of data coming from the Nephrology and Dialysis Unit of the Vigevano Hospital. The data set comprises 45 different patients with more that 200 dialysis sessions for each patient and with 10 different monitored signals (the time series features of Table 1) for each session.

As a first example, we have considered a case (patient #5, dialysis #72) in which, even though the outcome is classified as succesfull (i.e. type 1), a more subtle analysis reveals some sub-optimal behaviors in the monitored parameters.

In particular, the patient suffers from hypertension. Hypertension, in turn, may cause alterations in the hematic volume (HV) reduction. In a good session HV fits a model where, after a short period of exponential decrease, a linear decrease follows; hypertension may inhibite the exponential pattern, and lead to a slower reduction of the HV, that fits a linear model since the beginning of the session. As a matter of fact, in the case at hand this situation holds. Figure 3 shows on the left (first two columns) all the signals of the query case (patient #5, dialysis #72). Figure 4 (always on the left) highlights on the diastolic pressure (DP) and systolic pressure (SP) as well as on the HV.

Retrieval was performed by asking for a high similarity (distance threshold equal to 0.15) with respect to DP, SP, blood bulk flow (QB) and HV; QB is the first shown signal of the cases in figure 3 and has been considered as an important contextual factor of the retrieval. In correspondence to these very relevant features, we also set the highest weights to be used for global similarity calculation.

The right part of figure 3 shows an overview of the first retrieved case (patient #5, dialysis #36), while figure 4 (on the right) details the situation of the DP, SP and HV features.

These time series behaviors look very similar in the two cases (see figure 4): in particular, hypertension is present in both situations; moreover, even though less data points are available for the query case with respect to the retrieved one, it is clear that the HV decreases linearly, missing the initial exponential pattern.

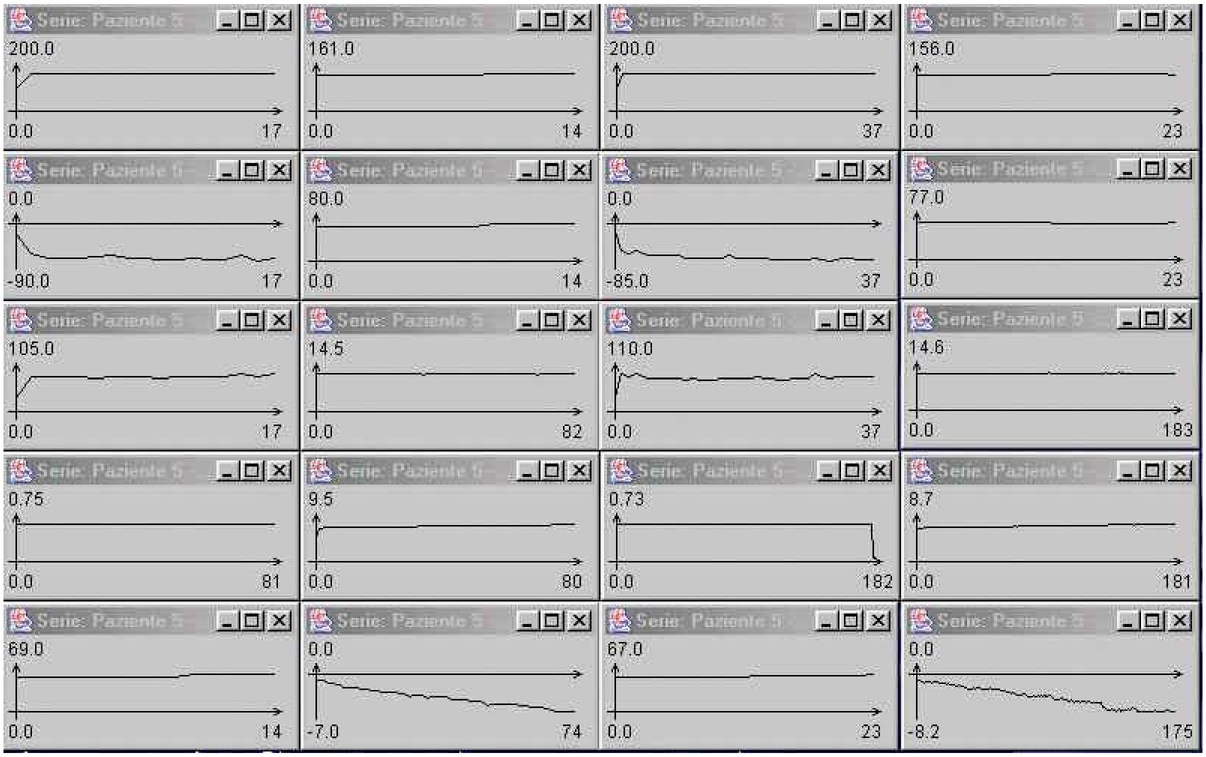

Figure 3: Example of retrieval of signals of a dialysis session: query case (on the left), best retrieved case (on the right). Numbers on the ordinate of each graphic represent maximum or minimum value of the time series.

Observe that also the retrieved case was labeled as successful by the physician. Actually, it seems that the outcome definition is based just on a macroscopic observation of (a subset of) the features. On the other hand, our system allows to obtain a deeper insight of the situation, highlighting types of anomalies which, if they don't lead to an immediate dialysis failure, could produce poor therapeutic results in the long run.

As a second example, we have considered a case (patient #10, dialysis #71) in which some alterations in the extra-corporeal blood circuit took place. This kind of problems (typically due to an occlusion of the patient's fistulae) are indicated by a sudden increase of the arterious pressure (AP) around the end of the session, and by a corresponding decrease of the venous pressure (VP). Retrieval has been conducted by requiring a high similarity for AP and VP, and by assigning them the highest weights. Figure 5 details the values of AP and VP for the query case and for the best retrieved one; the overview is not provided due to lack of space.

Observe that, while the query case was labeled as succesfull (type 1 outcome), the retrieved case has a type 2 outcome (i.e. succesfull after nurse intervention). In particular, the nurse provided the patient with a diuretic drug, to compensate hypotension.

This result has led us to consider the values of DP and SP (see figure 6): as a matter of fact, the values are low in both cases (in particular, the final increase in the retrieved

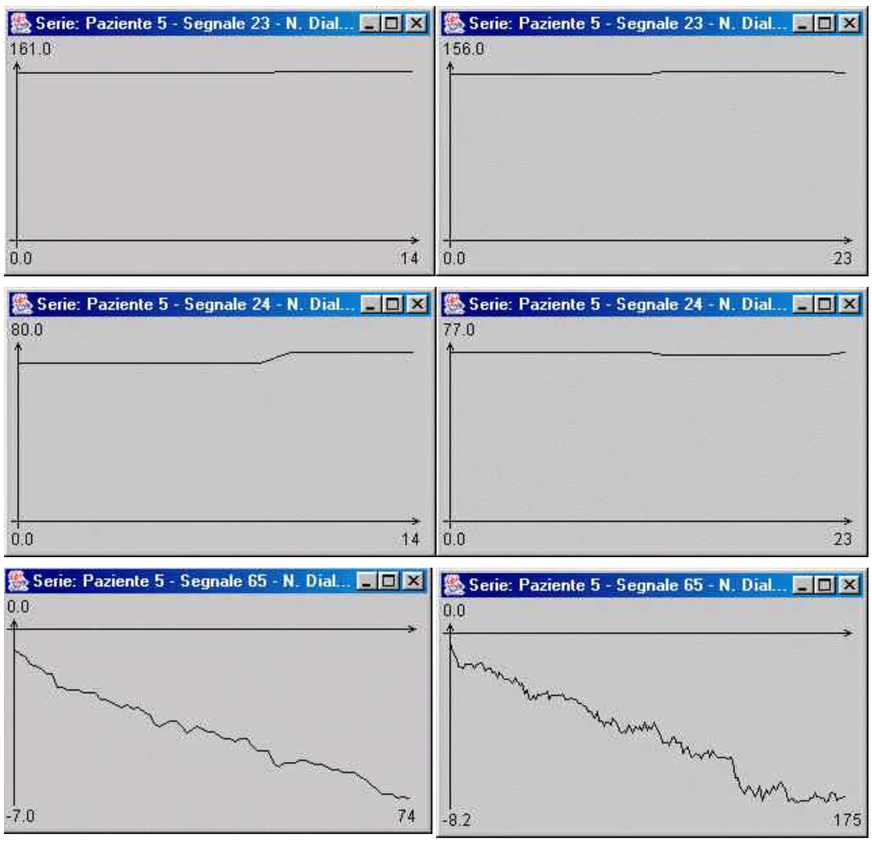

Figure 4: Diastolic, systolic pressure and hematic volume retrieval (cfr. fig. 3).

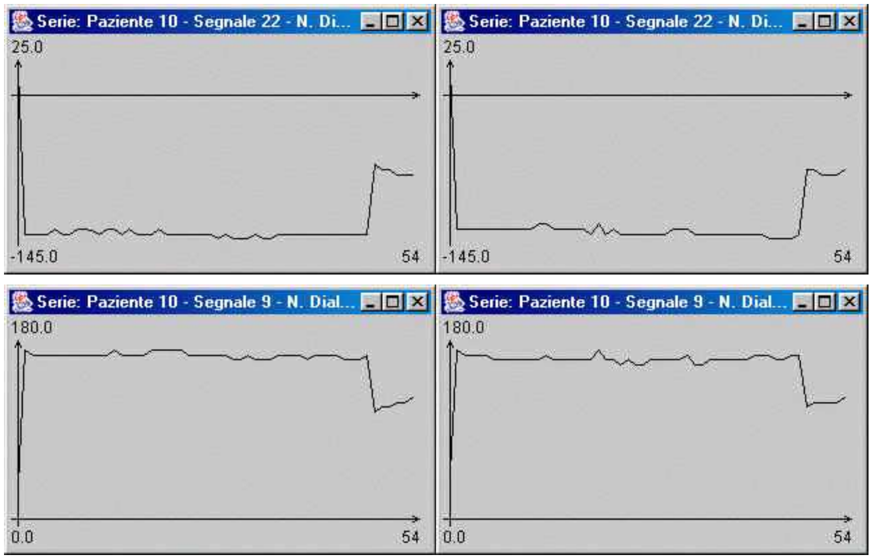

Figure 5: Arterious and venous pressure retrieval in the second example.

case corresponds to the drug effect). The information provided by the retrieval procedure can thus warn the physician to pay particular attention to hypotension for this patient, since in the past a medical intervention was required, in a situation that is extremely similar to the current one.

In conclusion, our tool provides results that allow to better assess the dialisys efficiency, and that can indicate directions for further analyses and considerations.

4 Conclusions and Future Works

In this paper, we have described an application of case-based retrieval in a time-dependent clinical domain: the treatment of ESRD patients. Despite only the first phase of the CBR cycle is implemented, the system architecture is non-trivial, as retrieval is articulated as a multi-step procedure. Moreover, since most of the case features are in the form of time series, dimensionality reduction (based on DFT) and indexing techniques (based on k-d trees) have been relied upon.

The system is provided with a user-friendly graphical interface, which allows the physician to tune the retrieval parameters (e.g. ranges and importance of the features), in order to focus her/his attention on different aspects of the dialysis sessions. Moreover, s/he can choose whether to visualize the overall case structure, or to concentrate the retrieval on a single feature. In this way, the tool proves to be a flexible means for realizing an explorative analysis of the patient's data: it allows to look for similar situations (typically patterns corresponding to persistent failures over time), understanding if these patterns are repeated over the same patient or over different ones, and retrieving what solutions have been provided in those cases, in terms of dialysis prescription (i.e. the prescribed flow rates at the beginning of dialysis). This information can be adopted by the physician

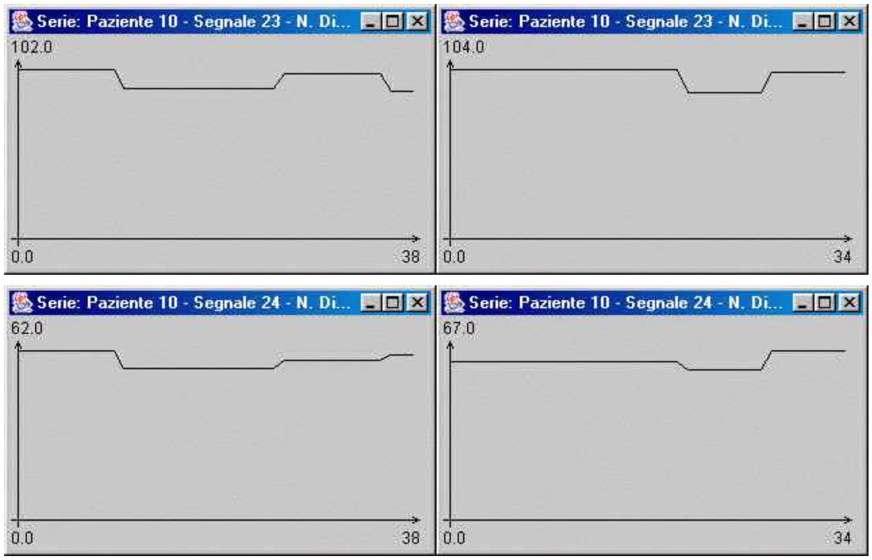

Figure 6: Diastolic and systolic pressure retrieval in the second example.

to characterize the current patient and to identify the best therapy adjustments to be implemented.

Moreover, the system could be relied upon for quality assessment, i.e. to assess the performance of the overall hemodialysis service at hand and to isolate the reasons of failures. Technically speaking, quality assessment requires to fulfil two tasks: (1) retrieve similar time series within the process data, in order to assess the frequency of particular patterns, (2) discover relationships between the time patterns of the process data and the performance outcomes. Clearly, our tool would be suitable for task (1), but it could also be embedded within a more complex tool, able to summarize the dialysis sessions from a clinical quality viewpoint (see e.g. [3]).

The system version described in this paper is still a prototype, that retrieves the data from ad hoc files. From the technical viewpoint, in the future we plan to interface it with a commercial DBMS. In this way, the DBMS into which dialysis variables are stored by the hemodialyzer would directly be used as the case repository, making the system easy to be integrated into clinical practice.

At present, we have implemented dimensionality reduction through DFT and we resort to a k-d tree as an index structure where range queries can be directly performed. We are currently studying the possibility of adopting alternative methods such as Discrete Wavelets Transform (DWT) [4] or Piecewise Constant Approximation (PCA) [8, 9].

Moreover, we are also evaluating the option of substituting the k-d tree index structure with TV-trees [13], an organization able to efficiently access data in very large dimensional spaces (this would allow us to resort to a larger number of coefficients to represent a time series, thus speeding up retrieval[4]).

TV-trees are based on the idea that, if lots of elements all agree on certain attributes,

[4]Postprocessing time may be significantly reduced.

then the index has to be organized by branching on those attributes themselves. An efficient algorithm for k-NN queries on TV-trees is described in [13].

Note that performing a k-NN query (and thus providing only the parameter k) is more intuitive for a physician with respect to working with range queries. As a matter of fact, in this case a range (a number between 0 and 1) has to be specified for *each* feature, and range values don't have an immediate mapping to the physical interpretation of the features themselves. The request to specify the ranges from one side allows a fine tuning of the retrieval results, but on the other hand sometimes forces the physician to make several tests before finding a really suitable value, that guarantees a non empty intersection of the different query results.

Finally, we plan to make an extensive testing of our approach, working on new real patients' data coming from the Nephrology and Dialysis Unit of the Vigevano Hospital in Italy. The retrieval system will also be integrated as a data inspection facility within a commercial tool currently deployed at Vigevano, that supports physicians in the on-line management of dialysis sessions.

Acknowledgments

We are very grateful to Dr. Roberto Bellazzi of the Dialysis Unit of the Vigevano Hospital for having made available the set of cases and for fruitful discussions about the topics discussed in this paper.

References

[1] A. Aamodt and E. Plaza. Case-based reasoning: Foundational issues, methodological variations and system approaches. *AI Communications*, 7(1):39–59, 1994.

[2] R. Agrawal, C. Faloutsos, and A.N. Swami. Efficient similarity search in sequence databases. In D. Lomet, editor, *Proc. 4th Int. Conf. of Foundations of Data Organization and Algorithms*, pages 69–84. Springer-Verlag, 1993.

[3] R. Bellazzi, C. Larizza, P. Magni, and R. Bellazzi. Quality assessment of dialysis services through temporal data mining. In *Proceedings 9th Conference on Artificial Intelligence in Medicine Europe*, Cyprus, 2003.

[4] K.P. Chan and A.W.C. Fu. Efficient time series matching by wavelets. In *Proc. ICDE 99*, pages 126–133, 1999.

[5] R. Van der Spek and A. Spijkervet. Knowledge management: dealing intelligently with knowledge. In L.C. Wilcox J. Liebowitz, editor, *Knowledge Mangement and its Integrative Elements*. CRC Press, 1997.

[6] M.L. Hetland. A survey of recent methods for efficient retrieval of similar time sequences. In H. Bunke M. Last, A. Kandel, editor, *Data Mining in Time Series Databases*. World Scientific, 2003.

[7] D.B. Leake J.L. Kolodner. A tutorial introduction to CBR. In *Case Based Reasoning: Experiences, Lessons and Future Directions*, pages 31–65. AAAI Press, 1996.

[8] E. Keogh. A fast and robust method for pattern matching in time series databases. In *Proc. Int. Conf. on Tools with Artificial Intelligence*, 1997.

[9] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and Information Systems*, 3(3):263–286, 2000.

[10] A.V. Oppenheim and R.W. Shafer. *Digital signal processing.* Prentice-Hall, 1975.

[11] R. Schmidt, S. Montani, R. Belazzi, L. Portinale, and L. Gierl. Case-Based Reasoning for medical knowldge-based systems. *International Journal of Medical Informatics*, 64:355–367, 2001.

[12] M. Stefanelli. The socio-organizational age of artificial intelligence in medicine. *Artificial Intelligence in Medicine*, 23(1):25–47, 2001.

[13] V.S. Subrahmanian. *Principles of Multimedia Database Systems.* Morgan Kaufmann, 1998.

[14] D.R. Wilson and T.R. Martinez. Improved heterogeneous distance functions. *Journal of Artificial Intelligence Research*, 6:1–34, 1997.