From Texts, Images, and Data to Attribute Based Case Representation

Michael M. Richter[1] and Rosina O. Weber[2]

[1] University of Calgary, Calgary, Canada
mrichter@ucalgary.ca
[2] Drexel iSchool, Philadelphia, USA
rosina@drexel.edu

Abstract. In this article we study complex case representations in Case-Based Reasoning. To some degree this is a survey paper. But in addition it gives a unified approach to solving the problems connected with representations mentioned in the title in a way that has not been considered so far. The most popular form to represent cases use attribute-based representations. They allow an easy formulation of similarity measures and retrieval functions. However, in practical applications, case problems and solutions are in the first place given in other ways, e.g. by using texts, images, sensor data or speech data. On this level it is hard to apply reasoning and in particular CBR. This is due to the difficulty to determine similarity measures and retrieval functions. In order to overcome this we introduce a general level structure that allows to bridge the gap between bit-oriented low level and the attribute-oriented high level that is accessible to humans as well as CBR systems. The approach is put in the form of a process model.

Keywords: Case-based reasoning, process model, text, image, sensor data, speech, similarity, retrieval.

1 Introduction

There are quite many different case representations in Case-Based Reasoning (CBR). The most popular ones are attribute-based representations. However, information both for queries and solutions can be represented in quite different ways. Major ones are texts, images and sensor data, which includes speech data. There are several problems connected with these representations, mainly with respect to similarity definitions and retrieval techniques. For that reason one wants to transform such representations into one that is based on features (that we use as synonym to attributes). In addition often it is not easy to grasp the information content what asks again for a transformation. For CBR applications we present a uniform way to do this.

The transformations are applied to representations on different levels of abstraction. The lowest level is the one on which objects are represented in a computer. It goes up to the level of understanding and where traditional CBR methods can operate. The representations mentioned have several characteristics in common:

- No precise semantics is given.

- Humans are sometimes but not always able to extract useful information immediately.
- There are different methods available to manipulate the objects at the different levels.
- Most of the commonly used similarity measures cannot be used.
- The retrieval is not simple and the commonly used functions do not apply.

These characteristics prevent us from using CBR directly as in attribute-value based representations. Hence one has the desire to convert such representations into ones that are based on attributes. In fact, many have been the attempts for this and many are quite successful. However, all these approaches have been kind of ad-hoc. In this paper we present a first approach towards a unifying principle in the form of a process model. Such models turned out to be quite useful. The most prominent example is the CBR process cycle; see [Aamodt-Plaza 1994].

Our model introduces a level structure of different levels of abstraction. The levels reach from a low bit level in which objects are stored in the computer up to high levels understandable by humans that can be handled by a standard CBR system. These levels are connected by computable functions so that one can switch between them. These are described by a connection graph. For reaching a representation accessible to a CBR system or one suitable for human understanding one needs to select a path in the graph. Here one needs in general to go up as well as down in the levels abstraction. In addition we will briefly discuss similarity measures and retrieval functions in the context of this model.

Much of the content of this paper will appear in a forthcoming book "Case-Based Reasoning: a textbook" by Michael M. Richter and Rosina O. Weber, [Richter, Weber 2012].

2 The Different Case Representations in the View of CBR

First we will discuss essential properties of different representation methods. As mentioned, attribute based representations are the most popular. In the simplest form they are just vectors indexed by attributes. Often, however problems and solutions are not represented in this way. We consider three other representation types: Texts, images and sensor data including spoken information. First we will briefly discuss their purpose and the problems for a CBR approach.
All representations have more or less some structure as shown in Fig. 1. On the left side the structure is completely hidden and the more one moves to the right the structure becomes visible.

Fig. 1 Structuring

2.1 Text

Text is in natural language; it contains its ambiguities, redundancy, and impreciseness. These elements provide essential difficulties to automatically manipulating text. Texts can be parts of a query as well as of a solution. When aiming to use text for problem solving, text needs to be understood, for formulating the query as well as for making use of the solution.

We refer to text as more or less structured collections of words in a well-known language that convey a meaning when interpreted in aggregation. These collections are not organized in meaningful representations like attribute-value pairs.

Sometimes, texts are semi structured, like in emails or hypertext when some machine distinguishable features such as date and subject are also available. Other variation of semi-structured text is when sub headings are given to paragraphs or, in some way, we can determine the topic of a given set of words.

Text understanding can be done by humans and machines. If it is done by humans then they can reformulate it in a structured way so it is more easily handled by machines. To better describe the exact challenges to understanding and consequently on the exact problems Textual CBR methods need to overcome, Fig. 1 contextualizes text occurrences in a spectrum moving from unstructured towards structured forms through levels of growing availability of machine recognizable symbols that inform meaning. Note that this spectrum denotes a continuum as perceived by machines. See different approaches to move in this sprectrum in [Brüninghaus, Ashley 2005] and [Asiimwe et al. 2007].

At the very left in Fig. 1 is text that has not received any kind of processing, it is merely a collection of alphanumeric symbols like letters that makes explicitly no sense.

The similarity between the problem and textual cases should indicate the extent texts have knowledge of interest to the problem (because this is its utility). The problem can be partially represented in text, or cases are documents and the case base is a document collection. For retrieval a similarity measure is needed that compares problems and documents or documents and documents.

A specific problem comes up when both query and answer uses texts. For example, “Which Wikipedia page is closest to my master’s thesis?”

Here one has to compare two texts as a whole what necessitates difficult preparatory work.

2.2 Images

The saying “An image says more than a thousand words” puts forth the idea that images carry a lot of information and can make them easily accessible. Images are utilized in different disciplines for different purposes. Images can occur in queries as well as in solutions. We restrict our discussion here to 2-D images.

a) Images can be a part of the query asking:
   - Is this part of the body healthy and if not show me the pathological parts?
   - Is there a swimming pool on this aerial view?
   - Which image in the case base is most similar to the presented one?

b) An image Im may be part of an answer further contributing to its solution. Such answers could be:
   - An image shows a specific cancer type.
   - The image shows a typical shape of a station wagon.
   - The image shows an aerial view of alpine mountains.

In addition, images may be mixed with attribute values, both in the query and the answer. For instance, “Show me an aerial view with a swimming pool of size larger than X." A CBR application using images in medicine is given in [Grimness, Aamodt 1996].

2.3 Sensor data

Sensor data arise from measurements as in medicine or in natural sciences and engineering. These measurements often record continuous data. A special situation is in speech understanding; see [Rabiner, Juang 1993]. For the relation to CBR see [Maier, Moore 2009]. Speech reaches the human ear in a wave form. They can contain much information. A basic problem for these representations is to formally understand them. Otherwise one cannot meaningful describe queries and answers.
Typical examples are:
Temporal measurements on patients: Do they indicate a pathology?

(1) In mining: Is the air pressure coming to a dangerous point?
(2) We would like to see a typical measurement of a pathological situation.
(3) A person is speaking in a noisy environment. What did the person say?

For simplicity we restrict ourselves to linearly ordered measurements and we interpret this ordering as time. An application in medicine is shown in [Xiong, Funk 2007].

3 The Local-Global Principle

In order to formulate a systematic approach we use a general structural representation principle. It is based on the view that (complex) objects to be compared are built up in a systematic way starting from elementary (atomic) elements. The objects can be thought as very general ones like machines, the human body, images, etc.

This systematic structure is called the local-global principle for complex object descriptions; it says:

- There are elementary (local or atomic) description elements; for simplicity, we assume that these are attributes.
- Each object or concept A is (globally) described by some construction operator C from the local elements:

$$A = C(A_i \mid i \in I).$$

Here I is some index set for the atomic elements A_i (i.e., the attributes). The construction operator can generate a flat vector description or it can be a very complex construction, for instance describing a complex machine or building using nested graphs and trees.

This principle is not only applicable to objects occurring in a case description. It has a wide variety of other uses and it will be used for describing similarity measures. The local-global principle gives rise to two tasks:

The decomposition task. Break the object or concept down into atomic parts. Often, this task happens because an object may be presented globally and the parts are initially unknown. Popular examples occur when one is faced with complex goals. Such goals have to be decomposed in order to fulfill them.

The synthesis task. Compose an object or concept from simpler parts. This occurs when one wants to construct a complex object like a machine or building. However, it plays also a role for getting a similarity measure.

Both tasks play a role for relating the concepts we are interested in. The principle allows also to annotate parts if the constructed object is analyzed further, for instance with respect to importance or danger.

For the objects under investigation, the local-global principle has very different realizations. The claim is not that the principle itself is very innovative, in fact, it is quite standard. The point is the unified use of the principle in order to allow a systematic treatment of the different techniques.

In addition, the principle allows dealing with informal description elements. Then the constructed object will not be precisely given. This occurs frequently in everyday life and can suffice to solve problems.

4 The Process Model and the Level Structure

4.1. The Process Model

When complex case representations are considered the problem starts with the interpretation of the term reasoning in CBR. In logic and in logic oriented Artificial Intelligence one performs reasoning by applying certain rules. This is standard in the context of attributes. In CBR this is replaced by searching for a solution of a problem using similarity measures: Instead of giving a precise and correct answer the similarity selects an approximate candidate for the solution. The difficulty for these representations is that one cannot use them directly for CBR and similarity reasoning.

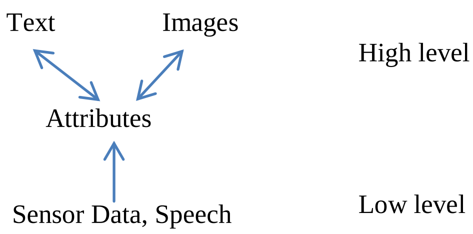

Fig. 2 Relating the representations

The basic idea is to convert such representations into one that is based on attributes as shown in Fig.2. This has to be done in a systematic way. For this, the transformations have to be much more detailed using methods from the specific areas.

All these representation types can be used to build cases. They can describe queries as well as solutions. But only attributes can be used for reasoning. For that reason a conversion is of interest. Such steps run under the name feature extraction.

The necessary methods for the transformation are organized in a directed connection graph.

Definition 1: The connection graph has

1) Nodes as data structures, and
2) Edges that are labeled by the conversion methods.

It is important to remark that that the edges do not always go up in the abstraction hierarchy. Now we define a central notion.

Definition 2: The process model refines the structure in Fig. 2 and is defined in short in three steps:

1) Definition of the levels structure and explain their using.

2) Definition of computational methods that connect the levels.
3) Establishing and analyzing the connection graph.

The process model realizes the local-global structure in a specific way. It constitutes a uniform way for approaching the problems associated to the different presentations. Levels are already known to us as for instance from object oriented programming. Here we follow this principle but from a more radical way. The hierarchy represents different forms of abstraction.

For bridging the gaps between the different levels there are many tools and methods available that should be stored in case bases. This has the advantage for the user that no knowledge about the internal structure of the method is needed. For calling the method one has simply to know its purpose. There is a rich collection that we cannot cover here and we give only a brief listing.

In the connection graph data structures on the different levels are connected by the methods. The methods and the graph will be usually discussed together.

The goal of analyzing the connection graph is to find a path to an endpoint with a data structure of interest, for instance one that is accepted and available for the purpose of interest. In many situations it is an attribute-value representation. It depends on the problem and the solution which direction from low or high or vice versa one has to go in order reach such a representation. If the problem is presented as a complex object like a text or an image often one has to decompose it in order to find the solution. If elementary data are given the synthesis has to be performed.

As in every process model the realizations of the levels and the methods can be quite different. In the sequel we will define these levels and the needed methods for texts, images, and measured data. The specific levels depend on the chosen representation method and to some degree on the intended application. The hierarchy represents different forms of abstraction.

The lowest level is given by the representation in a computer. This is always a form of representation in bits. Although it describes the objects completely it is of little use for humans. Our brain does not perceive the elementary details of such a representation but performs automatically an abstraction to higher levels. This gives rise to the task to do this abstraction by a computer so the computer can do more advanced reasoning

This is done stepwise. Each step has to be made by a computational method. The first level where humans have an intuitive access to is the level of known objects, often a domain oriented level. The objects here are familiar ones like parts of the brain or lung in medicine or parts of a machine in engineering.

We call the highest level the overall level. Here one is concerned with aspects of interest. On this level one discusses domain objects, certain relations between them or the values of parameters associated to them. The discussion is concerned with tasks like diagnosis, planning, design, etc. This is analog to what one does when reasoning with attributes.

The levels are not totally separated. For operating on a certain level one sometimes needs access to methods defined on lower levels.

In principle, the levels are organized for bridging the gap between the low level descriptions represented in a computer and the high level semantics in which humans are interested.

The general level structure is shown in Fig. 3.

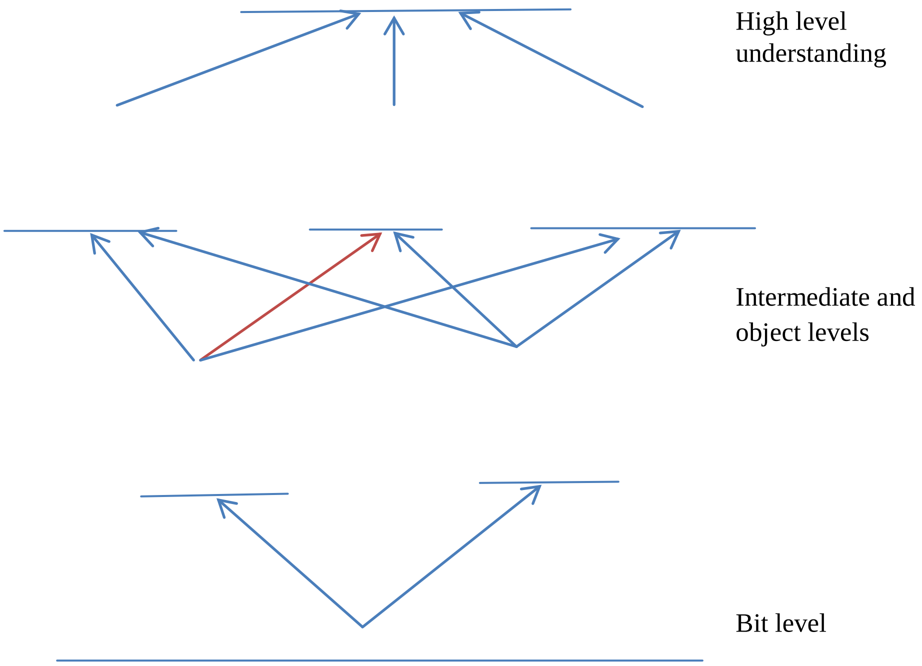

Fig. 3. The general level structure

For the decomposition task one goes down from the highest level (for instance when a problem is formulated in a human oriented way) to the lower levels accessible to the computer.
For the synthesis task one goes in the opposite direction. This happens for instance if parts of an object are defined on a low level that have to be combined to an object on a higher level on which its usefulness can be described.

4.2 The Level structure

The level structure for each of the complex representations we discuss here vary. In the next sections we will discuss each of them and then describe the methods and connection graphs to each.

The Level structure for Texts. An overview over the text levels is depicted in Fig.4.

1) The most elementary level is the level of alphanumeric symbols. These have no meaning unless they are used for denoting words that have a definition.
2) In the level of words**:** They are built up by alphanumeric symbols (letters). They have a meaning and often more than one. The orthography describes the correctness of the words.
3) The level of relations with words is the simplest level where words are combined. Phrases follow. They are not simply sets of words; their ordering and grammar are important. FAQ are examples of structures that combine and relate multiple phrases.

4) Hypertext structures are semiformal representations of sentences. They contain formal as well as informal elements. Hypertexts organize (informal) text objects in a (formal) network that are connected by edges called hyperlinks.

Complexity	Levels	Examples/structure
Simple	**Alphanumeric symbols**	Keywords, Attributes
	Words	Lexicons
	Relations between words	Sets of words Thesauri Glossary
	Phrases	Sentences, Headings
	FAQs	Question-answers
	Hypertext	Cross links between words in sentences
Complex	**Free text**	Multiple paragraphs

Fig. 4. Text levels

5) The most complex level is the one of sentences. They combine words and phrases using grammatical rules. The sentences are elements formulated in a language. The grammar describes the correctness of the sentences in that language. However, understanding their meaning is much more complex. The grammatical rules of a language vary in complexity.
6) The free text is the highest level for presenting sentences. Usually, understanding requires knowledge that is not explicitly mentioned. That makes automated understanding difficult.

The problem is that the descriptions on the most abstract level containing free text cannot be used for CBR. For this reason one needs to introduce the attribute level as shown Fig. 5.

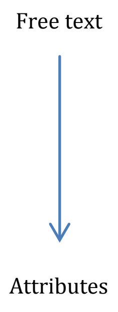

Fig. 5. Text attributes

The Methods and the Connection Graph for Texts. The traditional method to process an unstructured text sequence in natural language processing is syntactic analysis. The output is a representation structure that informs the roles of each word in the sequence. Identifying the functions of words helps understanding, adds knowledge and structure, helping further manipulation for reasoning. See also [Mott et al. 2005].

When using natural language processing (NLP), generally three steps are considered. The first is pre-processing. Second is part-of-speech (POS) tagging. POS tagging assigns a part-of-speech to each word (or token) in the text. The part-of-speech consists of a grammatical function of the token in the text. The last step is parsing, that builds the resulting representation: a parse tree.

Bag of words (BOW) is the simplest form where a set of words is tokenized without keeping their original ordering. This can be implemented multiple ways, as in or without preprocessing steps like removing stop words or using a stemmer. Typically, bag-of-words are used in vector models.

Converting text to hypertext, what is called post hoc authoring, requires division of the original text into meaningful units as well as meaningful interconnections of the units. This division is the result of text segmentation.

The feature extraction connection the text to the attributes is the most problematic method. Feature extraction is a learning method that aims at capturing a set of features that is sufficiently representative that it works as an effective model of the original. The goal of feature extraction in CBR is to identify indices for guiding retrieval. Therefore, the same characteristics we expect of a good index can be used as reference criteria for feature extraction. One of the strategies used in feature extraction is to use information gain as a criterion to identify discriminatory features. For more see [Wiratunga et al. 2006].

For comparing two texts as a whole additional work has to be done. Instead of extracting features from one text, one can search for words, sequences, and models that both texts t1 and t2 have in common. This can be done in different ways, and is commonly mentioned as co-occurrence:

Find structures w that occur both in t1 and t2.

Co-occurrence is a method to compare two texts. For more see [Weeds, David 2005]. Tools are also available, see for example [FREETEXTTABLE].

The Level Structure for Images. The computer cannot immediately operate at the highest level of abstraction when confronted with an image. In order to empower the computer one has to interpret an image, i.e. to associate meaning with an image. The hierarchical structure can represent different levels of abstraction for that purpose.

a) Lowest level: The pixel level. On this level the computer retrieves and maintains a digital image as an array of pixels. This level is analog to the alphanumeric level in textual representations. At this level, any knowledge contributing to the understanding of the image data is still hidden in primitive, numeric data types.
b) The geometric level: Here elementary geometric objects are introduced as lines, curves, areas with their boundaries and brightness, segments etc. and segments. These properties have parameters like size, length, diameter and visual properties such as hue, brightness, shadings, and textures. These are attributes of the geometric objects.
c) The symbolic level: The geometric objects also have a symbolic description what allows symbolic reasoning as well as CBR reasoning. It has to be taken into account that mathematical objects are mere approximations

of real objects. In addition they do not occur as exactly mathematical objects and they do not satisfy precisely the mathematical definition, some real-world occurrences are vaguely defined, as for instance “ovals” are informal variants of ellipses. However, CBR needs to deal with such representations.

d) The domain-specific level: This is the lowest level regarding the domain. The objects are defined either symbolically or geometrically. In medicine they are parts of the body, in an aerial view it may be a swimming pool in the landscape.

e) The overall description level: At this level the meaning of the image is formulated in such a way that it provides an answer to the original problem. In the overall description one is concerned with the domain objects in two ways:

1. Relations between two objects, for instance:
   a. Absolute positions of objects and their parts in the image as “in the upper left corner there is an object X”
   b. Relative positions: Object X is left of object
   c. Distances between objects.
2. Parameter values of objects, for instance, the size of objects.

The Methods and the Connection Graph for Images. A general recipe for creating the higher levels is provided as follows.

1) Apply mathematical functions at the given level, first on the low pixel level.
2) Apply methods that can be applied for obtaining a finite set of geometric objects.
3) Translate them into a symbolic description: Feature extraction.
4) Combine the descriptions in order to obtain a description that is meaningful in the context of the application domain.

There is a huge amount of methods operating on the low level and we mention some principle types only:

- Point operators: Operators that have point as arguments only and do not consider points in the neighborhood.
- Local operators: Have as arguments not only points but also points in the neighborhood. Such operators have a higher chance to grasp aspects of the meaning of the image.
- Global operations: Are not restricted to local sets of pixels.
- Segmentations: Special global operators that consider regions of the image, see [Frucci et al. 2008].
- Morphological operators: Transform partial objects into other ones more easily to view or to interpret such that the original part can be reconstructed. Important examples are simplifying operators as
  - The threshold operator omits all pixels with values above (or below) a certain threshold value.
  - The closing operator preserves the rough parts exactly. Small missing parts are added.
  - The opening operator preserves the rough parts in an approximate fashion, small details are omitted.

As said above, in CBR one would store such image processing methods in case bases.

The feature extraction goes in general in two steps:

1) Defining the geometric objects of interest selects parameters of interests for further steps from the geometric objects. That is those which are used in the description of the domain objects.
2) Selecting the needed parameters of these objects. A list of examples is in Table 1.

The combination into domain objects as well as the description of the overall level is standard symbolic operations.

Table 1. Geometric parameters

Name	Meaning	Computed
Gravity Center	Center of mass of a segment.	xy-point
Segmentation	Dividing the image into blocks or clusters of pixels based on the pixels' properties	
Simplifying operators	Deleting or adding details.	pixel data
Bounding box	Rectangle parallel to the *x-y* axes, describing the convex hull for an object. pixel clusters	geometry
Gray value	Mean gray value of an image or an image segment. Can be either in HU values or in absolute pixel values.	real value
Texture	Grey level differences as contrast, size of area where change occurs, directionality.	parametric features

The Level Structure for Sensor Data and Speech. Here we encounter several low levels with digitals as objects. A first difficulty is that the physical data do not arrive in the form of bits. Often, they arrive as continuous data. For this reason the first abstractions are defined on a very low level with discrete bits.

For simplicity we assume that the bits are linearly ordered. This ordering can be a temporal ordering what puts another structure on the signal. On the next level the time level is separated into segments.

The approach is to introduce a level structure starting from the signals to reaching the overall understanding at the highest level. This requires a deeper investigation of how the target objects are presented for comparisons.

For sensor data and speech the lowest level contains the signals provided by (possibly) stochastic processes. These data may be noisy, due to external or internal noise (for instance from the speaker).

For reaching the level that allows extracting features one looks at the signals as the very low level elements. For considering the stochastic processes we restrict the discussion to hidden Markov processes (HMMs). Therefore the approaches to these processes and dynamic situations are usually of interest. In general we have the following levels:

1) The level of the stochastic processes HMM: This is concerned with signals.
2) The level of the features: Here the signal sequence is represented formally and in a way that allows comparisons by similarity measures.

3) The language level. Here combinations of features are formed that are of interest in the applications and can be denoted by names. This is a level of symbolic representation.
4) The overall level: Here the meaning of the signal sequence is understood.

On the feature level there are only attribute-value vectors left. These are vectors of numbers with a fixed length. On this level the most adequate similarity measures operate.

The Methods and the Connection Graph for Images for Sensor Data and Speech. The main parts of the graph are shown in Fig.6 in a simplified way.

Fig. 6. The simplified connection graph

The initial methods are employed as preprocessing steps. They are of mathematical nature:

1) Removing noise.
2) Discretization.
3) Breaking the signal sequence into segments.
4) Extracting the speech features using extraction methods: They are the only elements that are furthermore considered and should therefore represent the information contained in the original data fairly well.

The first step is done by removing noise. Noise occurs in many applications as in a lab or when one directs a robot. If there is no noise or one uses a headset there are excellent tools as [Dragon 2006] or [IBM ViaVoice 2006].

What follows is a discretization of the continuous signal. This done by recording individual signals at selected points of time. As a result we obtain a sequence of bits.

Here we encounter a very large set of signals that are the result of a stochastic process. The stochastic process is usually non stationary what makes it difficult to handle. One should be aware of the fact that in speech one usually has 48000 signals per second.

In order to achieve a distribution that is close to stationary one breaks the time scale into smaller segments. In speech recognition the length of the segments is oriented on intervals in which the vocal tract producing the speech does not change its behavior.

After the preprocessing steps the most important one is feature extraction. An initial question is to define what the features for a time sequence should be. Today, one distinguishes mainly two types of features.

1) The first one is of statistical nature. The extracted features as mean, variance or correlation coefficients are sometimes called Meta data in statistics. They are of interest for statistical purposes.
2) Another principal and frequently used kind of methods uses representations of a signal sequence S(n) in terms of certain basic functions φ_t. Here we assume that the signal already occurs in discrete times n= 0,1.2….

The representation is

$$S(n) = \sum_{i} c_i \, \varphi_i(n)$$

Here the coefficients (or the first ones) serve as features.

A specific widely used instance of such representations is in terms of trigonometric functions. The most elementary situation is the representation as a Fourier series. In this case the transformation is called the Discrete Fourier Transform (DFT) of the signal. More details can be found in [Funk and Xiong 2007].

The feature vectors now represent the original signal sequence. They are the only objects that can be used for similarity.

This is in particular the case for the use of similarity measures. However, here are situations where additional problems occur when dealing with feature vectors. Two major ones are:

1) The vectors are ultimately produced by a stochastic process.
2) There is a connection between the individual signals due to the time ordering.

In principle, on can define similarity measures as well as retrieval methods taking place on all levels where the objects are represented. On some levels this cannot be done directly according to established methods and on other methods they are not of interest. For instance, one can define such measures on the bit level. This may be in the concern of Pattern Recognition but not of CBR. Also one cannot define meaningful measures directly on whole images. Our concern here is where two aspects can be combined: Computational efficiency and meaningful results for problem solving.

There are two main levels of interest:

a) The feature level.
b) The symbolic level.

As said, similarity measures on the lower levels are central to Pattern Recognition and are not our concern. If features are generated then similarity measures simfeat compare feature vectors fv:

$$\text{simfeat}\,(fv1, fv2) \in [0,1].$$

These similarity measures are in particular appropriate for comparing temporal measurements and speech recordings. Originally many data are represented as feature vectors of short length. They constitute the case base and can be elements of the queries too. One has to take care that all feature vectors are of the same length.

This has a direct application in speech recognition. One records a number of spoken words in the form of feature vectors in a case base. In order to find out which word w is spoken in an actual situation the feature vector of w is compared with all feature vectors in the case base and the nearest neighbor is taken.

Therefore the algorithm starts by mapping the first elements of each time series and then successively adds new mappings until all elements have been included. The function d(i,j) denotes the distance between the i-th value of the first time series and the j-th value of the second time series. It is usually taken as the difference between the costs arising for i and j.

Formally DTW is defined recursively as:

DTW(i,j) = d(i,j) + min{DTW(i-1,j), DTW(i,j-1), DTW(i-1,j-1)}

The initial value is DTW(0,0) = 0, the end is at i-1, j-1.

In this equation, DTW(i,j) is the minimal cumulative distance when mapping the first i values of the first time series to the first j values of the second time series. Since the order has to be preserved, there are three options to choose the next mapping:

1) Move to the next element in the first time series only;
2) Move to the next element in the second time series only;
3) Move to the next elements in both time series.

Thus the value DTW(i,j) is calculated as the sum of the distance d(i,j) and the minimal cumulative distance for a mapping that ends with (i−1,j), or (i−1,j-1), or (i,j−1). The steps are shown in Fig. 7.

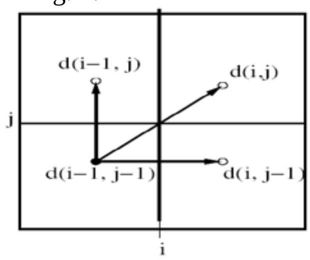

Fig. 7 DTW step

5 Conclusion

In this paper a unifying principle was formulated and detailed as a process model for describing the conversion of different case representations. The motivation was that texts, images and measured data provide several problems to be used in CBR systems. In contrast, attribute-value representations are easier to handle by standard CBR techniques. For this reason one wants to convert problematic descriptions into attribute-value based ones.

In addition, often one wants to convert the representation into another one where certain information is not hidden anymore. The process model supports this and does not consist of isolated and ad-hoc methods. Several details for applying the model for text, images and sensor data are provided.

Acknowledgements

Rosina Weber is supported in part by the U.S. EPA-Science to Achieve Results Program and the U.S. Department of Homeland Security Programs, Grant # R83236201.

References

[Aamodt, Plaza 1994] Aamodt A. & Plaza. E. (1994) Case-based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AICOM 7(1):39-59, 1994.

[Deller et a 2000] Deller, J.R.,Hansen, J.H.L., Proakis, J.G.: Discrete-Time Processing of Speech Signals. IEEE Press 2000.

[Dragon 2006] Naturally Speaking. http://www.dragonsys.com/, 2006.

[Funk, Xiong 2007] Funk,P., Xiong, N.: Extracting Knowledge from Sensor Signals for Case-Based Reasoning with Longitudinal Time Series Data. Case-Based Reasoning on Images and Signals (ed. Perner, P.) pp. 247-284. Springer Verlag 2007

[IBM ViaVoice 2006] http://www.4.ibm.com/software/speech/, 2006.

[Maier, Moore 2009] Maier,V., Moore, R.K.: The Case for Case-Based Automatic Speech Recognition, Proc. Interspeech 2009.

[Rabiner, Juang 1993] Rabiner, R.L., Juang, B.H.:, Fundamentals of Speech Recognition. Prentice-Hall Inc., 1993.

[Xiong, Funk 2007] Xiong,N., Funk, P.: Concise case indexing of time series in health care by means of key sequence discovery. Applied Intelligence 28, pp. 247-260. 2007.

[Frucci et al.2008] Frucci, M., Perner, P., di Baja, G.S.: Case-Based Reasoning for Image Segmentation. Case-Based Reasoning for Signals and Imaging (ed. Perner, P.):. Springer Verlag 2007.

[Grimness, Aamodt 1996] Grimnes, M., Aamodt, A.: A two layer case-based reasoning architecture for medical image understanding. Advances in Case-Based Reasoning. Smith, I., Faltings, B. (Eds.), Springer Verlag, pp164-178. 1996.

[Holt, Benwell 1996] Holt,A., Benwell, GL.: Case-Based Reasoning and Spatial Analysis. Journal of the Urban and Regional Information Systems Association, 8(1):27-36. 1996.

[Perner 2001] Perner, P.: Why Case-Based Reasoning Is Attractive for Image Interpretation. Proc. of the 4th International Conerence on Case-Based Reasoning pp.27-43, Springer Verlag 2001.

[v.Wangenheim 1996] von Wangenheim, A. : Cyclops: Ein Konfigurationsansatz zur Integration hybrider Systeme am Beispiel der Bildauswertung. Dissertation Kaiserslautern, 1996.

[Richter, Weber 2012]. In Press. Case-based Reasonin: a Texbook. Springer: Berlin.]

[Brüninghaus, Ashley 2005]Brüninghaus, S. & Ashley, K. D. 2005, “Reasoning with Textual Cases” in Muñoz-Avila, H. & Ricci, F.eds. Case-Based Reasoning Research and Development (LNAI 3620), Springer, Berlin, pp. 137-151.

[Mott et al. 2005] Mott, B., Lester, J. & Branting L. K. 2005, “The Role of Syntactic Analysis in Textual Case Retrieval” in Weber, R. & Branting, L.K. eds. Proceedings of the Textual Case-Based Reasoning Workshop, Chicago, IL, pp.120-127.

[Asiimwe et al. 2007] Stella Asiimwe, Susan Craw, Bruce Taylor, and Nirmalie Wiratunga Case Authoring: From Textual Reports to Knowledge-Rich Cases. In R.O. Weber and M.M. Richter (Eds.): ICCBR 2007, LNAI 4626, pp. 179193, 2007. Springer-Verlag Berlin Heidelberg 2007.

[FREETTEXTRTABLE] http://msdn.microsoft.com/en-us/library/ms177652.aspx.

[Weeds, David 2005] Weeds Julie and Weir David (2005) Co-occurrence Retrieval: a General Framework for Lexical Distributional Similarity. Computational Linguistics 31(4) 439-476.

[Wiratunga et al. 2006] Wiratunga, N., Massie, S., & Lothian, R. (2006). Unsupervised Textual Feature Selection. 8th European Conference on Case-Based Reasoning (ECCBR-06), Springer, pp 340-354.