We're All Going on a Summer Holiday*: An Exercise in Non-Cardinal Case Base Retrieval†

Hugh Osborne and Derek Bridge
Department of Computer Science
University of York
Heslington
York YO1 5DD
U.K.

Abstract. In this paper, we present *similarity metrics*, which provide a seamless integration of cardinal and non-cardinal similarity measures. We show that cardinal and non-cardinal ways of measuring similarity in case bases are simply instances of the definition of similarity metrics, and that a single definition of the maxima of a similarity metric can be applied in both the cardinal and non-cardinal cases, to give intuitively sensible maxima in both cases. We present a number of ways of constructing new similarity metrics from existing similarity metrics. A great strength of the framework is that similarity metrics of different types (e.g. cardinal and non-cardinal) can be combined, and the result will still be a similarity metric (to which the definition of maxima is still applicable).

The paper is illustrated throughout by examples from a holiday case base.

1 Introduction

Traditionally cases have been retrieved from case bases using weighted similarity measures, in which the similarity of a new problem to a case in the case base is given some numeric value [3, 16, 5, 6]. In [9, 8, 10] we suggested an alternative approach, in which a similarity measure would return a partial order on cases, in which a case c_1 is better than case c_2 if c_1 is more similar to the given seed than c_2. In order to contrast this approach to the traditional approach — which we called the *cardinal* approach [15] — we called this approach *ordinal*. In [9, 8, 10] we also discussed how the results from our ordinal approach might also be applied to the more traditional cardinal approach. While the ordinal approach is widely applicable, it does entail some loss of information. The resulting partial order might tell us that c_1 is better than c_2, but will not inform us *how much* better. Recently interest has also been shown in similarity measures that are neither boolean- nor numeric-valued [4, 13, 12]. In this paper we will demonstrate a new approach, called *similarity metrics*, which not only combines the wide applicability of the ordinal approach with the greater informational content of the cardinal approach, but also maintains and provides information from similarity measures of other types.

*Thanks to Cliff and the Shadows [14].

†To appear in the proceedings of the Sixth Scandinavian Conference on Artificial Intelligence, SCAI'97

The paper is intended to complement our paper at ICCBR [7]. In [7] we concentrate more on the theoretical background of similarity metrics, while this paper demonstrates more clearly how a case base retrieval system using similarity metrics might be implemented. This paper also extends the work presented in [7]. In particular, the identity metric (section 3.2.1), feature structure metrics (section 3.2.5), seeded similarity metrics (section 3.3), and the use of left composition to obtain "deep" features from "surface" features and of right composition to give "interval" metrics (both in section 3.4.4) are not covered at all in [7].

This paper will be illustrated throughout by examples from the "travel agent's" case base [1] maintained at the AI-CBR web site [2] at the University of Salford. This is a case base containing almost 1500 cases, each representing a possible holiday. We will demonstrate how cases can be selected from the case base by comparison of (combinations of) features, e.g. price, destination etc., and how retrieval will not only rank the cases, but also provide information indicating the degree of similarity of cases. For the sake of this paper we will use a small subset of this case base in our examples. This subset is given in the appendix. The reader is referred to the appendix for details of the structure of a typical case.

2 Approaches to Case Base Retrieval

2.1 Cardinal Measures

Obviously some of the features of the cases are amenable to the cardinal approach. For example the similarity of the prices of two holidays can be measured by their absolute difference. With slightly more difficulty, the similarity of months can be measured by their proximity — e.g. the similarity of November and December is 1, while that of August and November is 3. However, similarity of some of the features can only be mapped to a cardinal measure with great difficulty, if at all. Assume, for example, that the possible modes of transport are by car, plane, train or coach, and that the customer prefers to drive, and hates flying. What is then a reasonable numeric measure of the similarity of travel by train and by plane, and how does this relate to the similarity of travel by car and by coach? Similarly, what would be a useful cardinal measure of the similarity of a lakeside holiday in the mountains to an urban holiday at the seaside?

2.2 Ordinal Measures

In [9, 8, 10] we suggested how *ordinal* measures could be used to capture non-cardinal information. For example, when comparing modes of transport a user-defined ordering on these means could be defined as shown in Fig. 1. As discussed in [9, 8, 10], it is also possible to model cardinal measures using ordinal measures by simply ignoring the additional cardinal information — e.g. one price might be better than another if it is lower, the amount by which it is lower being considered irrelevant. Note that this entails some loss of information.

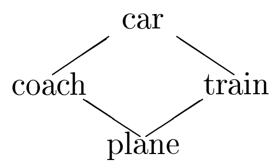

Figure 1: A user-defined ordering for modes of transport.

2.3 Other Types of Similarity Measure

The ordinal approach could also be used for comparing the activities available on the various holidays. The activities are given as a set-valued feature. These sets might then be compared using the usual subset ordering. Again this similarity measure might tell us that one holiday is better than another, i.e. it offers a superset of all the activities of the inferior holiday, but not indicating what these additional activities are. A better approach would, by analogy with the cardinal approach, not return simply a boolean value, but the set difference of two sets of activities, thus conveying information about the degree to which they differ.

3 Similarity Metrics: A Unified Approach

The three types of similarity measure discussed above all return some measure of similarity — in the ordinal case, simply a truth value; in the cardinal case some numeric indication of similarity; and in the set-valued case, a set of values by which one case can be said to “exceed” another. We have defined the notion of *similarity metrics* to provide a framework for integrating all of these types of measures, and many others besides. This work is intended to provide a sound theoretical basis for an algebra of case base retrieval, in a manner analogous to the rôle played by relational algebras in database retrieval. It is beyond the scope of this paper to give a full formal presentation of similarity metrics. The reader is referred to [7, 11] for a discussion of the formal aspects of similarity metrics.

Briefly, a similarity metric is a generalisation of a similarity measure. Where an ordinal similarity measure will take two cases and return a boolean value, and a cardinal measure will take two cases and return a numeric value, a similarity metric will take two cases and return some value from some ordered set[1]. This provides a powerful framework within which similarity metrics, possibly of differing types, can be combined to give new similarity metrics — for example two boolean valued similarity metrics can be combined to give a similarity metric returning pairs of boolean values, and an ordinal metric can be combined with a cardinal metric to give a metric returning a boolean-numeric pair. Similarity metrics can also be combined with other functions, allowing tranformations on data structures before and after application of the similarity metric. In such a way, for example, a cardinal similarity metric can be transformed to an ordinal metric, a similarity metric applicable to some individual feature of a case may be applied to complete cases, or disjunction or conjunction can be used to transform a similarity metric returning pairs of boolean values to one returning single boolean values.

We are now ready to define similarity metrics, and the maxima of a similarity metric.

[1] Actually we require this ordered set to be a complete lattice. See [7, 11] for the details.

This will be done in Sect. 3.1. We will then give examples of some simple similarity metrics in Sect. 3.2, illustrating them with applications to the example case base. In Sect. 3.3 we will introduce *seeded similarity metrics*, and again provide illustrations from the case base. Section 3.4 will show how to construct more complex similarity metrics, and illustrate this again in the example case base. All the illustrations are taken from a prototype implementation of a similarity metric based case base retrieval system implemented at the University of York.

3.1 Similarity Metrics and their Maxima

Definition 1 *A similarity metric is a pair* $(\trianglelefteq, \mathcal{L})$*, where* $\mathcal{L}$ *is a complete lattice, and* $\trianglelefteq$ *is a function from pairs of values from some set* A *to* $\mathcal{L}$*.*

As mentioned above, the reader can understand this paper by remembering that a complete lattice is an ordered set, the ordering being denoted $\sqsubseteq$ below; the other properties of a complete lattice are not directly appealed to in this paper.

The function $\trianglelefteq$ measures the relative values of its arguments — if $x \trianglelefteq y = v$ then y is said to *exceed* x by v, or, alternatively, v is the *excess* of y over x. Since $\mathcal{L}$ is a lattice, excesses need not be comparable.

We can define the concept of maxima for similarity metrics. A first approximation, by analogy with maxima for partial orders, would be to define the maxima to be those elements that always exceed any other element more than that element exceeds the "maximal" element — i.e. to define the maxima to be

Definition 2

$$\mathsf{max}\ S = \{x \in S | \forall y \in S : x \trianglelefteq y \sqsubseteq_{\mathcal{L}} y \trianglelefteq x\} \ . \tag{1}$$

However, $x \trianglelefteq y$ need not be comparable to $y \trianglelefteq x$, and this needs to captured in the definition of the maxima.

An element x of a set S is a maximum if, whenever the excess of x over y is comparable to the excess of y over x, then the excess of x over y is greater than or equal to the excess of y over x. An equivalent formulation is: if the excess of x over y is less than or equal to the excess of y over x, then the two excesses are equal.

Definition 3 *The maxima of a set* S *by a metric* $(\trianglelefteq, \mathcal{L})$ *are defined to be:*

$$\mathsf{max}\ S = \{x \in S | \forall y \in S : y \trianglelefteq x \sqsubseteq_{\mathcal{L}} x \trianglelefteq y \Rightarrow y \trianglelefteq x = x \trianglelefteq y\} \ . \tag{2}$$

3.2 Some Simple Similarity Metrics

3.2.1 Identity.

If A is itself a lattice a trivial, but useful, similarity metric is given by $x \trianglelefteq y = y$. The maxima of this similarity metric will be the maxima under the ordering on A. This can be shown by substituting the definition of this metric in (2):

$$\begin{aligned} \mathsf{max}\ S &= \{x \in S | \forall y \in S : y \trianglelefteq x \sqsubseteq_{\mathcal{L}} x \trianglelefteq y \Rightarrow y \trianglelefteq x = x \trianglelefteq y\} \\ &= \{x \in S | \forall y \in S : x \sqsubseteq_{\mathcal{L}} y \Rightarrow x = y\} \ . \end{aligned}$$

For example, the accommodation available is one to five star hotels, or holiday flats. The obvious lattice can be defined, with hotels being ranked according to the number of stars, and holiday flats being incomparable to hotels, so that, for example, $* \sqsubseteq **$, but $* \not\sqsubseteq$ Flat and Flat $\not\sqsubseteq *$. The similarity metric defined above will then give $* \trianglelefteq ** = **$ — i.e. $**$ exceeds $*$ by $**$, and $** \trianglelefteq *$ — i.e. $*$ exceeds $**$ by $*$.

In the prototype implementation at the University of York the retrieval request[2] `id_m acc_l` will select holidays 188, 680 and 1088, since 680 and 1088 are at four star hotels (there are no holidays in five star hotels available), and 188 is in a holiday flat (and therefore incomparable to all other holidays, which are in hotels).

3.2.2 Orders.

An alternative metric for an ordered set $(A, \sqsubseteq)$ is $(\sqsubseteq, \mathbf{Bool})$, where **Bool** is the usual boolean lattice. Again, the maxima are those of the ordering.

$$
\begin{aligned}
\mathsf{max}\ S &= \{x \in S | \forall y \in S : (y \sqsubseteq x \Rightarrow \sqsubseteq y) \Rightarrow (x \sqsubseteq y = y \sqsubseteq x)\} \\
&= \{x \in S | \forall y \in S : (y \sqsubseteq x \Rightarrow x \sqsubseteq) \Rightarrow (x \sqsubseteq y \Leftrightarrow y \sqsubseteq x)\} \\
&= \{x \in S | \forall y \in S : (y \sqsubseteq x \Rightarrow x \sqsubseteq y) \Rightarrow (x \sqsubseteq y \Rightarrow y \sqsubseteq x)\} \\
&= \{x \in S | \forall y \in S : x \sqsubseteq y \Rightarrow y \sqsubseteq x\}\ .
\end{aligned}
$$

Taking the user-defined order given in Fig. 1 to be `transPO`, the holidays with preferred modes of transport can be retrieved using `order_m transPO`. In this case, the excess of, for example, travel by car over travel by coach is *True*, while the excess of travel by coach over travel by train is *False*[3]. The holidays selected are those with identifiers 188, 370, 952, 1088 and 1184, since these all use travel by car, the preferred means of transport.

3.2.3 Numeric Values.

If $x \dot{-} y$ is the function on some numeric set $\mathbf{N}$ that returns the difference of x and y if x is greater than y, and 0 otherwise, then $(\dot{-}^{-1}, \mathbf{N})$ is a similarity metric. The inverse, $\dot{-}^{-1}$, is taken to give an intuitive reading to the metric, e.g. $3 \dot{-}^{-1} 7 = 4$ reads as "7 exceeds 3 by 4". The maxima of a set will again be the maxima given by the usual numeric ordering.

$$
\begin{aligned}
\max S &= \{x \in S | \forall y \in S : y \dot{-}^{-1} x \leq x \dot{-}^{-1} y \Rightarrow y \dot{-}^{-1} x = x \dot{-}^{-1} y\} \\
&= \{x \in S | \forall y \in S : x \dot{-} y \leq y \dot{-} x \Rightarrow x \dot{-} y = y \dot{-} x\} \\
&= \{x \in S | \forall y \in S : x \leq y \Rightarrow x = y\}\ .
\end{aligned}
$$

The holidays with the longest duration can be selected using `num_m` on the duration feature, which will select (only) holiday 188, since this is the only three week holiday.

[2] Actually, this similarity metric, and all others in this section, will have to be composed with a projection function to make it applicable to the whole case base. Composition of similarity metrics with other functions will be discussed in Sect. 3.4.

[3] In the case of this (boolean) metric a simpler paraphrase is possible. If x exceeds y by *True*, we simply say that x exceeds y.

3.2.4 Sets.

A similar approach can be taken with sets, using set difference. Again the inverse is taken to match an intuitive reading. The metric is $(\setminus^{-1}, \mathcal{P}(S))$, with $\mathcal{P}(S)$ the usual lattice on the power set of S. In this metric, for example, $\{a,b,c\} \setminus^{-1} \{a,c,d\} = \{d\}$ is pronounced "$\{a,c,d\}$ exceeds $\{a,b,c\}$ by $\{d\}$". Similarly, $\{a,b,c\}$ exceeds $\{a,c,d\}$ by $\{b\}$. This similarity metric will also give the usual maxima of the subset ordering.

$$\begin{aligned}
\max S &= \{x \in S | \forall y \in S : y \setminus^{-1} x \subseteq x \setminus^{-1} y \Rightarrow y \setminus^{-1} x = x \setminus^{-1} y\} \\
&= \{x \in S | \forall y \in S : x \setminus y \subseteq y \setminus x \Rightarrow x \setminus y = y \setminus x\} \\
&= \{x \in S | \forall y \in S : x \setminus y = \emptyset \Rightarrow y \setminus x = \emptyset\} \\
&= \{x \in S | \forall y \in S : x \subseteq y \Rightarrow y \subseteq x\} \\
&= \{x \in S | \forall y \in S : x \subseteq y \Rightarrow x = y\} \ .
\end{aligned}$$

For example, the similarity metric `set_m` applied to the activities feature will give the excess of {bathing, walking, diving} over {walking, diving, surfing} as {bathing}, and {walking, diving, surfing} will exceed {bathing, walking, diving} by {surfing}. This request will select those holidays offering a set of activities not subsumed by any other holiday. In the example case base, these will be holidays 132, 188, 370, 1184 and 1360. Holiday 1184 is the only holiday to offer ski-ing. The other three provide both bathing and walking, and subsume holidays 272 and 680, which only offer bathing, and holidays 711, 952 and 1088, which only offer walking.

3.2.5 Other Types.

Similarity metrics can also be defined for many other types — for example intervals, feature structures, graphs etc. Recent work by Jantke [4] and Plaza [12, 13] has proposed the use of feature structures as a case representation with the similarity between two feature structures being computed as the anti-unification. For example $\begin{bmatrix}\text{F a}\\ \text{G b}\end{bmatrix}$ and $\begin{bmatrix}\text{F a}\\ \text{H c}\end{bmatrix}$ would receive $[\text{F a}]$, their anti-unification — the information they have in common — as a similarity score. We can define a similarity metric that captures these ideas too. We would use the inverse of the subsumption ordering on feature structures as the lattice, and would defined a form of feature structure difference. Then, e.g. $\begin{bmatrix}\text{F a}\\ \text{G b}\end{bmatrix} \trianglelefteq \begin{bmatrix}\text{F a}\\ \text{H c}\end{bmatrix} = [\text{H c}]$ and this would be higher in the subsumption lattice than $\begin{bmatrix}\text{F a}\\ \text{G b}\end{bmatrix} \trianglelefteq [\text{F a}] = [\ \ ]$.

3.3 Seeded Similarity Metrics

The similarity metrics above are all "fixed" — i.e. they compute the similarity of two cases against some fixed metric, they cannot compute the relative similarity of two cases to some third case. This is similar to the rôle played by the "underlying partial orders" in our presentation of ordinal similarity measures [9, 8, 10].

As in ordinal similarity measures, we can generate new similarity metrics from existing similarity metrics, by means of standard constructions, in which the new metric computes the relative similarity of cases to some fixed case, known as the *seed.* To define these measures we introduce the concepts of *lesser distance* and *greater distance.* The lesser distance between two cases, in some metric, is the common ground of the excesses between the two cases — i.e. the greatest lower bound of the excesses. Similarly, the greater distance is the least upper bound of the excesses. So, for example, the lesser distance of two sets will be the intersection of the two sets, and the greater distance will be the union. If there is also a similarity metric defined on the result type of the (lesser or greater) distance a distance *relative* to some seed ϑ can be defined to be the excess of the distances of the cases from the seed.

So, for example, if the customer is looking for a holiday for 3 people, the most suitable holidays will be selected by[4] `rel_g_dist num_m (m_inv num_m) 3`. This similarity metric will compute the greater distance of its two arguments to the seed (3) — i.e. the absolute difference of the arguments to 3 — and then use the standard inverse numeric metric to rank these results, ranking lower excesses as better than higher excesses. So, in this metric, 2 exceeds 4 by 0 (since the absolute difference of both 2 and 4 with 3 is 0), and 6 exceeds 1 by 1. Applying this similarity metric to the example case base, there are two holidays that match the seed perfectly — 132 and 1360 — and these will be selected by this query.

3.4 Constructing More Complex Similarity Metrics

3.4.1 Inverses.

There are various ways of constructing new similarity metrics. It is possible, as mentioned in Sect. 3.3, to take the inverse of a metric, defined by $x \trianglelefteq^{-1} y = y \trianglelefteq x$. The obvious example for an inverse is the price, since most people will, within the bounds of other constraints, wish to minimise the cost of their holiday. The cheapest holiday will be selected by `m_inv num_m` applied to the price of the holidays, and this will, indeed, select holiday 1184.

3.4.2 Products.

The product of two or more similarity metrics may also be taken. These metrics are then applicable to tuples of values, and the result is an element of the product of the lattices in the original metrics. Say, for example, that the customer would like the best possible accommodation, at the lowest possible price. We can then use the product of the accommodation similarity metric and the price similarity metric `(id_m acc_l) m_prod (m_inv num_m)`. This metric selects holidays 188, 1088 and 1184. Holiday 188 is the cheapest (indeed the only) holiday flat holiday. Holiday 1088 is the cheaper of the two holidays in four star hotels, and holiday 1184 is not only the cheapest holiday, but also in a three star hotel — in other words, there is no point in presenting any of the two star hotels.

Note that the product of two similarity metrics, even when their result types are different, is still a similarity metric.

---

[4] This example makes use of an inverse similarity metric, in order to find cases with a minimum distance from the seed, rather than a maximum. Similarity metric inverses will be discussed in Sect. 3.4.

3.4.3 Prioritisations.

Inverses and products are the similarity metric counterparts to the standard lattice inverse and product operators. We have also defined *prioritisation*. The prioritisation of one metric over another will, conceptually, apply the first metric to make a first selection, and only apply the second metric to distinguish between cases that are considered equally acceptable by the first metric. For example, it may be that the customer does not want to compare the cheapest holidays in each class of accommodation, but definitely wants the accommodation of as high a quality as possible, and only wants to use the price to make a further selection. This can be achieved by taking the prioritisation of accommodation over price `(id_m acc_l) m_prio (m_inv num_m)`. In contrast to the product similarity metric, this similarity metric will only select holidays 188 and 1088, and not holiday 1184. Holiday 1184 may be cheaper than holiday 1088, but the accommodation is of an inferior quality.

3.4.4 Composition.

Similarity metrics can also be composed with other functions, allowing values to be transformed into a format more suitable for further computation. Both left composition (pre-processing) and right composition (post-processing) are possible.

Left Composition. In left composition the function being composed with the similarity metric is applied to the arguments of the similarity metric before the similarity metric is applied — i.e. it provides a pre-processing capability.

The most obvious form of pre-processing, which has been applied implicitly up to now, is projection, which will select some particular feature(s) of a case for comparison. So, for example, the cheapest holidays are actually selected by `(m_inv num_m) l_comp proj_price`. We will, in the remainder of this paper, again take the projections as assumed.

More complex pre-processing is, of course, possible. As the case base stands, it is difficult to generate a metric for selecting on grounds of the location. However, the case base [1] defines not only the locations, but also certain (geographical) features of locations — e.g. that the location is in the mountains, a lake, at the seaside. Locations can also be placed in a wider geographical context — e.g. that Holland is on the North Sea, Harz is in Germany and Fanø is in Denmark. Using this information, a function can be defined that, given a location, returns a set of descriptors of that location — so that, for example, a description of Lake Garda might be {Lake Garda, Alps, Countryside, Italy, Lake, Mountains, Waters}. We can apply this function in pre-processing and then use the standard set similarity metric in further selection. In this case, simply taking the maxima is unlikely to be very selective, since few descriptions of locations are going to be comparable. However the selection can be made more selective by applying seeded similarity metrics, as described in Sect. 3.3, to try and match holidays to some description of an ideal holiday given by the customer. Assume, for example, that the customer would ideally like a lakeside mountain holiday in Germany. A suitable query would then be `(rel_l_dist (id_m set_l) set_m {Germany,Lake,Mountains}) m_l_comp desc`. The second half of this expression, `desc`, will apply the description function to give a description of this location. The first half will then attempt to find those descriptions most closely matching the customer's ideal of {Germany, Lake, Mountains}. As it happens, there is no holiday perfectly matching the ideal, and the system recommends holiday 370 — at Lake Garda, offering lakes and mountains — and holiday 952. Holiday 952 is at Harz, the description of which is {Harz, Countryside, Germany, Mountains}, and therefore offers both "Germany" and mountains. No other holidays come as close to matching the customer's wishes.

Right Composition. In right composition the function being composed with the similarity metric is applied to the *result* of the similarity metric — i.e. it provides a form of post-processing.

As an example of post-processing, let us again consider the problem of selecting by price. We have shown how to select the cheapest holiday by means of `num_m`. However, a more likely scenario is that the customer says "I'm looking for a holiday at around DM 1400". A first approximation is to use a seeded similarity metric `rel_g_dist num_m (m_inv num_m) 1400`. However, this is going to be too selective, returning the one holiday with its price closest to DM 1400 — in this case holiday 952, at DM 1482. We can use right composition to make the second half of this similarity metric less selective. By applying an integer divide, by, for example, 500, we can map the distances computed to equivalence classes in multiples of 500, so that, if the difference in price of two holidays is less than DM 500 their relative distance is 0, and if the difference is, for example, DM 1350 their relative distance is 2. This gives us the similarity metric `rel_g_dist num_m ((div 500) r_comp (m_inv num_m)) 1400`. This will, indeed, select holidays 132, 370, 952 and 1088, which are the only holidays priced within DM 500 of DM 1400.

We would like to emphasise again that all these ways of combining similarity metrics from existing metrics remain within the framework, so that, for example, the maxima can still be taken. We consider this to be a great strength of the framework.

4 Conclusions

We have presented an integrated framework for computing many different types of similarity. The framework not only subsumes many different types of similarity measure, it also allows different types of similarity measure to be combined, while remaining within the framework.

The framework also allows an elegant formulation of pre-processing of input and post-processing of output, again while remaining within the framework.

We intend to investigate the combination of similarity metrics with relational algebra, leading to a *relational metric algebra* — a framework in which both database and case-base operations can be specified and reasoned about.

5 Acknowledgements

Dear All,
Having a lovely time in Helsinki. Thanks for all your help.
Wish you were here.
Hugh and Derek

The Intelligent Systems Group
Department of Computer Science
University of York
Heslington
York YO1 5DD

Appendix: The Example Case Base

Id	Type	Price	Nº of People	Location	Trans.	Days	Mon	Acc.	Hotel
132	{Bathing, Walking}	1147	3	Baltic Sea	Train	7	Sep	**	Hotel am Leuchtturm, Warnemunde
188	{Bathing, Walking}	2187	2	Atlantic	Car	21	Aug	Flat	Holiday Flat Atlantik, France
272	{Bathing}	3198	2	Corfu	Plane	14	Sep	***	Hotel Mirabell, Corfu
370	{Bathing, Walking}	1608	2	Lake Garda	Car	14	May	**	Hotel Palme, Lake Garda
680	{Bathing}	3644	4	Mallorca	Plane	7	Jul	****	Anlage Lago Playa, Mallorca
711	{Walking}	2274	4	Giant Mountains	Coach	14	Jun	**	Hotel Bedrichov, Giant Mountains
952	{Walking}	1482	4	Harz	Car	7	Jan	**	Hotel Fürstenhöh, Harz
1088	{Walking}	978	2	Tyrol	Car	7	Jan	****	Hotel Alpenland, Tyrol
1184	{Ski-ing}	508	2	Tyrol	Car	7	Jan	***	Hotel Adler, South Tyrol
1360	{Bathing, Walking}	4708	3	Fuerteventura	Plane	14	Mar	***	Hotel Crystal Beach, Fuerteventura

This is a small subset of the case base [1] at the University of Salford. In the original case base the second feature (the type of the holiday) is not, as here, a set-valued feature. The set values here have been adapted from the original case base to simplify the presentation of set-valued similarity metrics.

References

[1] AI-CBR Case-Base Archive. `http://146.87.176.38/cases.htm`.

[2] AI-CBR Home Page. `http://www.salford.ac.uk/docs/depts/survey/staff/IWatson/cbr01.htm`.

[3] R. Bareiss, J.A. King, J. Ashley, K. Kolodner, B. Porter, and P. Thagard. Panel on "similarity metrics". In *Proceedings of DARPA Case Based Reasoning Workshop*, pages 66–84. Morgan Kaufmann, 1989.

[4] K.P. Jantke. Nonstandard concepts of similarity in case-based reasoning. In H.H. Bock, W. Lenski, and M.M. Richter, editors, *Proceedings der Jahrestagung der Gesellschaft für Klassifikation,* Information Systems and Data Analysis: Prospects, Foundations, Applications, pages 29–44. Springer Verlag, 1994.

[5] P. Klahr and G. Vrooman. Commercialising case based reasoning technology. In I.M. Graham and R.W. Milne, editors, *Research and Development in Expert Systems VIII*, pages 18–24. Cmabridge University Press, 1991.

[6] J.L. Kolodner. *Case Based Reasoning.* Morgan Kaufmann, 1993.

[7] Hugh Osborne and Derek Bridge. Similarity metrics: A formal unification of cardinal and non-cardinal similarity measures. To appear in *Proceedings of the 2nd International Conference on Case-Based Reasoning,* ICCBR'97.

[8] Hugh Osborne and Derek Bridge. A case base similarity framework. In Ian Smith and Boi Faltings, editors, *Advances in Case-Based Reasoning,* Proceedings of EWCBR'96, pages 309–323, 1996.

[9] Hugh Osborne and Derek Bridge. Parallel retrieval form case bases. In Ian D. Watson, editor, *Proceedings of the 2nd UK CBR Workshop*, pages 43–54, 1996.

[10] Hugh Osborne and Derek Bridge. A formal analysis of case base retrieval. Technical Report YCS 281, Department of Computer Science, University of York, 1997.

[11] Hugh Osborne and Derek Bridge. Relational metrics: Integrating case based reasoning and database management systems. Technical report, Department of Computer Science, University of York, 1997. In preparation.

[12] E. Plaza. Cases as terms: A feature term approach to the structured representation of cases. In *Proceedings of the First International Conference on Case-Based Reasoning.* Springer Verlag, 1995.

[13] E. Plaza. On the importance of similitude: An entropy-based assessment. In Ian Smith and Boi Faltings, editors, *Advances in Case-Based Reasoning,* Proceedings of EWCBR'96, pages 324–338, 1996.

[14] Sir Cliff Richard. *Summer Holiday.* Columbia, 1963. (Columbia DB4977).

[15] M.M. Richter. Classification and learning of similarity measures. In *Proceedings der Jahrestagung der Gesellschaft für Klassifikation,* Studies in Classification, Data Analysis and Knowledge Organisation. Springer Verlag, 1992.

[16] M.M. Richter and S. Wess. Similarity, uncertainty and case-based reasoning in PATDEX. In R.S. Boyer, editor, *Automated Reasoning: Essays in Honour of Noody Bledsoe*, pages 249–265. Kluwer, 1991.