A Unifying Semantic Distance Model for Determining the Similarity of Attribute Values*

John F. Roddick[1] **Kathleen Hornsby**[2] **Denise de Vries**[1]

[1] School of Informatics and Engineering,
Flinders University of South Australia,
PO Box 2100, Adelaide 5001, South Australia.
Email: {roddick,denise.devries}@infoeng.flinders.edu.au

[2] National Centre for Geographic Information and Analysis,
University of Maine, Orono, Maine 04469-5711, USA.
Email: khornsby@spatial.maine.edu

Abstract

The relative difference between two data values is of interest in a number of application domains including temporal and spatial applications, schema versioning, data warehousing (particularly data preparation), internet searching, validation and error correction, and data mining. Moreover, consistency across systems in determining such distances and the robustness of such calculations is essential in some domains and useful in many. Despite this, there is no generally adopted approach to determining such distances and no accommodation of distance within SQL or any commercially available DBMS.

For non-numeric data values calculating the difference between values often requires application-specific support but even for numeric values the *practical* distance between two values may not simply be their numeric difference or Euclidean distance.

In this paper, a model of *semantic distance* is developed in which a graph-based approach is used to quantify the distance between two data values. The approach facilitates a notion of distance, both as a simple traversal distance and as weighted arcs. Transition costs, as an additional expense of passing through a node, are also accommodated. Furthermore, multiple distance measures can be incorporated and a method of 'localisation' is discussed which allows relevant information to take precedence over less relevant information. Some results from our investigations, including our SQL based implementation, are presented.

Keywords: Semantic distance, difference measures, similarity.

1 Introduction

In most applications, determining the relative distance between two objects through an inspection of the values of selected attributes is an important function. For simple numeric domains, this does not often cause a significant problem. However, for non-numeric or non-planar numeric domains, even those that are enumerated, this requires application-specific support. Despite this, there is no generally adopted approach to determining semantic distance and there is currently no accommodation of distance within SQL or any commercially available DBMS. We use the term *semantic distance* to refer to the notion of relative or useful (as opposed to lexicographical, linguistic or physical) distance between concepts.

The kinds of application requiring such support vary widely and include:

- temporal and spatial applications, in which the quickest route may not be the shortest or cheapest and vice versa,
- schema versioning, in which data stored under one protocol must be comparable with new data stored under a later protocol,
- data warehousing, in which the summarisation and cleaning of data may be achieved more efficiently through the clustering of objects with similar values,
- search engines, in which the entered keywords might only be indicative of the useful keywords to use when searching, i.e. a query using the keywords Venezuela and Duck might also be interested in articles which mention the Orinoco Goose,
- validation and error correction, where the closeness of an attribute's value to a predefined set of values may require checking and/or correction, and
- data mining, in which the proximity of objects or the extraction of rules about clustered objects may be required.

In general, each application that requires such support currently adopts one of two alternatives. Either the design and, more significantly, the population of comparison tables from scratch, or the forced translation of data with respect to some reference taxonomy. Although easier to implement, the latter option has the effect of losing the original descriptions, with the potential loss of useful information, while for the former, the compilation of distance tables may be expensive, particularly since the domain from which the attribute takes its values may be large. Moreover, the determination of distance may need to be calculated consistently across systems that may be controlled by different organisations. A commonly agreed and flexible policy is therefore required.

This paper proposes such a policy that can enable, as required, standardised distance datasets to be constructed, adopted and exchanged. The policy

Copyright ©2003, Australian Computer Society, Inc. This paper appeared in the Twenty-Sixth Australasian Computer Science Conference (ACSC2003), Adelaide, Australia. Conferences in Research and Practice in Information Technology, Vol. 16. Michael Oudshoorn, Ed. Reproduction for academic, not-for profit purposes permitted provided this text is included.

*Kathleen Hornsby's work is partially supported by a grant from the National Imagery and Mapping Agency, NMA201-00-1-2009.

is flexible, in that it subsumes many of the reported discussions on this problem[1], and practical, in so far as the computational complexity is low and simple modifications to query languages such as SQL can be implemented. The method for distance calculation developed in this paper allows for the reuse of semantic distance information. We also believe that many of the more advanced ideas being discussed in the literature regarding object clustering can also be accommodated. One advantage of a common model is the reusability of comparison tables and functions, a general agreement on the methods of applying semantic distance over a (group of) applications and ideally, as will be discussed later, enhancements to standards such as SQL.

The paper is organised as follows. The next section discusses in more detail the nature of distance and similarity. Following that in Section 3 we will develop our ideas of semantic distance and present a general method for its calculation. We also discuss the complexity of the method and, in Section 4, compare our model to those outlined in the literature. In Section 5 we discuss our work in implementing a system, including changes to SQL, that utilises these ideas. Finally, there is some discussion of future directions in Section 6 and a summary given in Section 7.

2 Conceptualising Distance and Similarity

There are many ways in which we can determine the semantic distance between two objects. For example, for numeric data, the data values can be viewed as direct arguments to a calculation function. For non-numeric data, some form of similarity procedure can be developed that correlates non-numeric instances with numeric values. Such methods commonly produce a numeric value indicating the closeness of the values according to some accepted convention and scale. However, in this respect, it can be argued that the reduction of non-numeric data to numeric proximity values can, for some applications, be improved, particularly when the value itself has no meaning except as a comparison.

Even for numeric data, in many cases the *useful distance* between two values may not simply be the numeric difference between them. A useful measure of distance for some spatial applications, for example, may be a measure of the *time* taken to get from point A to point B rather than any of the numerous methods of measuring physical distance. Importantly, this value may change as the mode of transport changes.

For temporal and geospatial applications, as well as in other conceptual spaces, the distance between two concepts[2] as viewed through one or more given models may be of interest. For example, the temporal interval concepts of *overlaps* and *meets* (*q.v.* (Allen 1983)) are closer in semantic terms than, say, the concepts of *overlaps* and *after*. Similar semantics exist for spatial terms (Kainz, Egenhofer & Greasley 1993).

The difference or similarity between two objects may also be a function of a number of attributes, each with different distance measures. For example, finding cities *similar* to Adelaide first requires each fact known about Adelaide to be compared to those of other cities *according to the semantics of that fact*, and then an aggregate difference determined between Adelaide and each city evaluated.

We can thus consider distance with reference to four forms of (non-orthogonal) abstraction:

1. numeric attribute values used directly,
2. non-numeric attribute values converted to numeric values for comparison,
3. distances between concepts / objects as represented through different models,
4. concepts / objects represented by aggregations or clusters of (numeric or non-numeric) values.

In considering how to calculate distance and similarity, there are a number of points which must be taken into account:

- In many cases, a simple numeric subtraction is either not possible or makes no sense. One street name cannot be subtracted from another and even house numbers provide a poor indication of the distance between residences. Even attribute values where subtraction seems suitable (such as grid references) may contain subtle problems on closer inspection. The calculation of distances around the earth, for example, requires the use of non-Euclidean mathematics[3].
- Psychological research indicates that relative measures are often more readily understood than absolute values, particularly if the value given to the distance would be arbitrary. Thus, in some cases, reverting to some numeric representation of distance may be inappropriate and notations indicating that *the distance between A and B is the same as between C and D* may be more convenient. For example, *the relative fall in share prices was similar to the magnitude of the fall in the October 91 crash.* That is, the focus may also be on the *changes* between relative distances rather than the relative distance itself.
- The model through which we view reality provides the measure(s) of distance. Arguably, few objective measures of distance exist and where they do, there are commonly multiple measures from which the modeller must pick one for use in the application. Colours, for example, may be divided by brightness, greyscale equivalence, colour disk distance, and so on, in order to determine the relative distance between them.
- The model of reality used also provides the measure of what is considered *sufficiently proximate* for a give use. Data warehousing and data summarisation techniques generally aggregate data to reduce storage space and provide a notion of *sameness* in the clustering of similar objects. The granularity of these values may be non-linear.
- Many applications, such as transport routes, classifications of diseases, and so on, utilise a model based on graphs rather than on simple hierarchies or trees. The distance following one path in a graph may be different from that for an alternative path.
- The semantic distance between two points is not always symmetric (*q.v.* (Rodríguez & Egenhofer 1999)). That is, the difference between A and B may not be the same as that between B and A.

[1] While not completely general, the authors could not find an application in the literature which was unable to be accommodated by this model.

[2] We use the terms *object* and *concept* in this paper in order to distinguish between the instantiation of a notion and its notion in a conceptual sense.

[3] This notion relates to Stevens' scales of measurements (nominal, ordinal, ratio, and so on.) where certain operations are not valid on, for example, nominal values (Stevens 1946).

- There may be a *transition* cost in passing through a node. For example, there may be a delay or some other penalty associated with passing through each town in a road trip. It may be appropriate to add a loading to the total distance for each intervening node to favour paths that pass through fewer nodes.

3 A Unifying Model for Semantic Distance

Our proposals aim to accommodate the issues outlined in the previous section in a general framework for determining the 'distance' between two objects in some context, while allowing applications that do not require the same semantic power to operate efficiently within a simplification of the same framework.

We propose a model as follows:

- For each domain requiring a distance measure (that is, for each different concept to be used), the values in the domain to be distinguished are arranged as a directed graph with the nodes representing points in conceptual space, and the arcs as connections between these points such that
  1. *distance* is a *useful*[4] semantic to use and,
  2. a numeric or non-numeric value, representing this distance, can be associated with each link.
- Nodes need not be named. However, external reference to unmarked nodes is prohibited.
- A value $d(n_i, n_j)$ representing the distance between each adjacent node is associated with each directed arc indicating the uni-directional or bi-directional distance between the nodes.
- Where numeric distances are used they need not imply meaning. However, some convention must be active such that the calculated values may be understood in terms of that concept. To ensure monotonically increasing distances, all distances must be non-negative.
- A distance combination function $\oplus$ must be specified or supplied. To ensure monotonically increasing distance, $\oplus$ is constrained to yield a value that is semantically no less distant than either of the arguments, ie $\forall i, j : i \oplus j \geq \max(i, j)$. $(i, j > 0)$. For most applications $\oplus$ will be simple addition but alternative methods of combining distances can be supplied, particularly for non-numeric values.
- A focussing (or zooming) factor ζ for the graph as a whole may be supplied to give a preference to concepts closer to the notion under consideration (*à la* (Hornsby & Egenhofer 1999)).
- Each node is given a (possibly zero) transition cost τ_{node} that is added to the cost of the path in cases where the node is neither the start nor end node.
- The distance $\mathcal{D}$ from a start node s to a final node f is computed as the minimum value of a function of the component distances d, as follows:

$$\begin{aligned}\mathcal{D}(s,s) &= 0\\ \mathcal{D}(s,f) &= d(s,f)\end{aligned}$$

  if s and f are adjacent

$$\begin{aligned}\mathcal{D}(s,f) = \min(&d(s,n_i) \oplus\\ &\zeta(\tau(n_i) \oplus \mathcal{D}(n_i,f)))\end{aligned}$$

  otherwise.

  where
  $d(n_i, n_j)$ is the distance between adjacent nodes n_i and n_j,
  ζ is the focussing or zooming ratio,
  τ is the transition cost of a node, and
  $\oplus$ is the combination function, (commonly arithmetic addition).

- Finally, a limit value $\mathcal{L}$ can be supplied above which any value of $\mathcal{D}(n_i, n_j)$ is considered infinite.

Thus for $\zeta > 1$ and $\tau \geq 0$, incorporating a greater number of steps has a penalty. For ζ less than one (and $\tau \geq 0$), objects reached through a higher number of nodes are advantaged (although for $\zeta > 0$ they still keep getting further distant). For example, the calculation of the shortest[5] distance between Navy and Orange in Figure 1(a) is affected by ζ and various values of τ, as shown in Figure 2.

If more than one distance measure is adopted, then additional labelled arcs can be included. In addition, a set of rules specifying the allowable combination of the different distance measures must be provided.

Note that the calculation proceeds recursively from the start node s and can thus be terminated if some maximum threshold is reached.

Consider the examples shown in Figure 1. In the graphs we have adopted zero as *no difference* and higher numbers as the magnitude of difference, ie. greater semantic distance. In some examples, only one (bi-directional) distance measure is included so arcs are labelled only with a single distance between nodes.

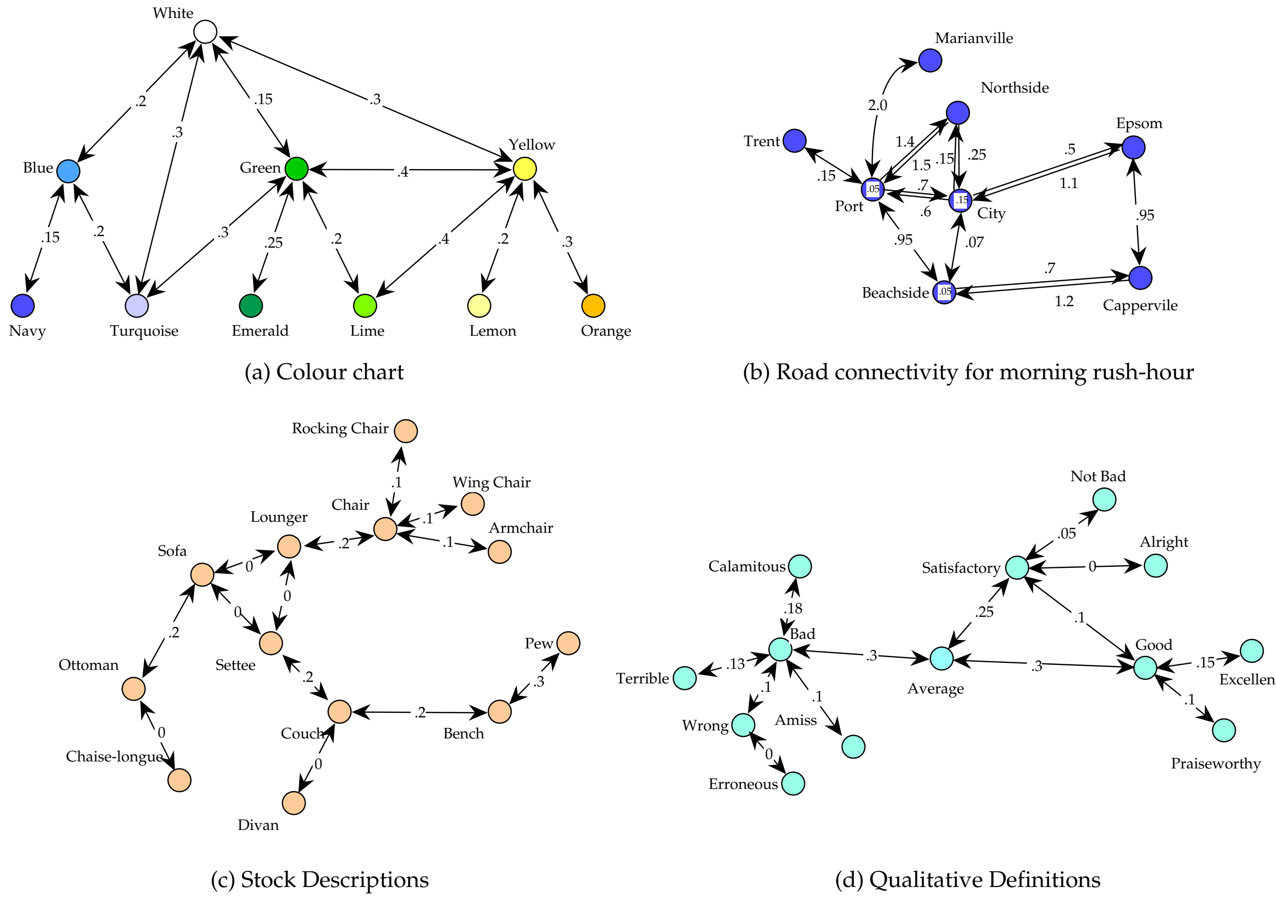
While the model is reasonably powerful, it can be simplified easily. Common simplifications include:

- Common distances on all arcs. Ie, a distance may be provided for the whole graph which is applied to all arcs. In this case the distance represents the number of arcs travelled.
- An assumption of symmetric distances (as indicated by bi-directional arcs). In this case, as in some of the examples in Figure 1, only one distance need be given.
- A tree structured graph. In this case, when coupled with common distances on arcs, the distance represents an indication of the height of the minimum unifying concept.
- A zooming factor of unity. In this case there is no penalty (or advantage) for being further away except by virtue of the addition of arc distances.
- A node transition cost of zero. That is, all costs are the result of traversing links.
- All link distances zero. That is, all costs are the result of passing through nodes.

If $\zeta \geq 0$ and either $\tau > 0$ or the distances in the graph are non-zero, the complexity of an algorithm (such as that in Figure 3) to compute the distance can be seen to be the same as the Shortest Path problem. Note also that, assuming again that $\zeta \geq 0$, the computation is deterministic for a finite number of nodes. Moreover, as the calculation is always additive, if the limit value $\mathcal{L}$ is provided (above which the concepts are considered *totally dissimilar*), the time to compute the distance can be further reduced.

[4] Usefulness is almost always contextual. However, many contexts share similar views of the distance between domain values and thus agreed values can often be determined.

[5] Note there are other paths which compute to higher values.



(a) Colour chart

(b) Road connectivity for morning rush-hour

(c) Stock Descriptions

(d) Qualitative Definitions

Figure 1: Example Applications

4 Related Work

Much of the work related to semantic distances relies on the linguistic or semantic similarity of terms that are based on a lexicographic definition of words or terms without reference to the application context. In this work we assume semantic similarity is dependent upon context. For example, work based on hierarchical schemata evaluates the similarity of terms based on spatial or mereological classifications (that is, part–whole relationships) of entities in which the granularity of the description of entities and their classes affects the value of the distance. In this case, the IS_A structure can add distance between similar concepts.

Kedad and Métais (1999) propose using metadata, including a linguistic dictionary, in a hierarchical structure with no distance set by the user. In their model, values are considered close if they belong to the same class. The semantic distance is fixed by the dictionary definition. They employ examples showing how classes of colours can be classified in terms of the various terms for shades of primary colours. The model proposed is then able, for example, to extract values similar to the colour *red* such as *vermilion* or *ruby*. The semantic similarity dependent upon a particular context is not readily extracted. For example, *pastel colours* would not be able to be processed as a query as it is not part of this colour hierarchy. In contrast, the work described here proposes a graph structure from which the distance between many conceptual classes can be determined.

Weinstein and Birmingham measure syntactic correspondence between definitions of pairs of terms. Their work deals with artificial ontologies and not real world complexities as *in the context of real-world applications, it is not possible to calculate the meaning of a term* (Weinstein & Birmingham 1999). They define concepts as having *roles* and *constraints*, where roles include a *relation*, ie. a relationship to other concepts. A concept that restricts the set of instances that can satisfy the relation is termed a *type restriction.* Contexts restrict accessibility within an ontological structure and are used to hide concepts and relations. Contexts are partially ordered and accessibility among contexts is transitive and non-symmetric.

> . . . in these structures, concepts are defined in relation to other concepts using logic. Local concepts inherit from shared concepts, and primitives are shared. In our artificial ontologies, concept definitions include roles with numeric and instance fillers, subject to unary and binary constraints. We . . . explore the nature of these structures: the degree to which we can predict overlap of concept denotations, and the potential usefulness of these predictions to support agent communication. (Weinstein & Birmingham 1999)

In our model, by having a separate concept graph for each context, real world complexities can be accommodated, because the context itself, defined by the user, gives meaning to the terms used.

Spanoudakis and Constantopoulos (1994, 1996) have investigated in depth the use of metrics to measure the distance between semantic descriptions of artefacts, particularly those developed at various stages of software development. Their model operates on semantic descriptions of objects using the modelling abstractions - *classification, generalization and attributes.* Objects are compared by four partial distance functions, which compare objects at different levels of detail, measuring *identification*, *classifica-*

	Steps						
	Navy → Blue	τ_{Blue}	Blue → White	τ_{White}	White → Yellow	τ_{Yellow}	Yellow → Orange
	0.15	0	0.2	0	0.3	0	0.3
ζ							
1	0.15		0.35		0.65		0.95
1.2	0.15		0.39		0.75		1.11
0.8	0.15		0.31		0.55		0.79

	Navy → Blue	τ_{Blue}	Blue → White	τ_{White}	White → Yellow	τ_{Yellow}	Yellow → Orange
	0	0.5	0	0.4	0	0.3	0.3
ζ							
1	0		0.5		0.9		1.5
1.2	0		0.6		1.08		1.8
0.8	0		0.4		0.72		1.2

	Navy → Blue	τ_{Blue}	Blue → White	τ_{White}	White → Yellow	τ_{Yellow}	Yellow → Orange
	0.15	0.5	0.2	0.4	0.3	0.3	0.3
ζ							
1	0.15		0.85		1.55		2.15
1.2	0.15		0.99		1.83		2.55
0.8	0.15		0.71		1.27		1.75

Figure 2: The effects of varying ζ and τ for the example in Figure 1(a).

```
Set solution (target value) to infinite.
Set calcdist (working value) to zero.
Non-deterministically follow all paths starting at s adding to calcdist until
   (a) f reached
      If calcdist < solution and calcdist < threshold then
         set solution to calcdist
      End-if
      Terminate search.
   (b) calcdist not less than solution
      more expensive path - Abandon search.
   (c) calcdist greater than threshold
      path too expensive - Abandon search.
   (d) node is marked
      looping - Abandon search.
If solution is infinite then no path has been found.
```

Figure 3: Marking algorithm for model

tion, *generalization* and *attribution* distances. The results of the partial distance function are aggregated into an *Overall Distance* measure which is then transformed into a *Similarity* measure. This model also introduces a *Salience* function, where salience is defined as the belief that an attribute is dominant based on a compound of the properties *charactericity, abstractness* and *determinance.* It is unclear, however, at what point the salience function is calculated and applied to the similarity measure.

Miller and Yang apply clustering techniques and a discrete distance function to measure distances over interval data where the interval distance measures the degree of association (Miller & Yang 1997). However, all examples shown are quantitative intervals. Their method assesses whether a *semantically meaningful distance metric is available* in order to *consider those attributes together and apply clustering to the set of attributes.* It is unclear how non-numeric intervals are treated.

Richardson and Smeaton combine the lexical database *WordNet* (Fellbaum 1998) with Resnick's measure of similarity to give a semantic similarity measure that can be used as an alternative to pattern matching (Richardson, Smeaton & Murphy 1994). They use *synsets* (synonymous word forms), collocations (connected words) and a hierarchical concept graph (HCG) with semantic pointers to hyponyms/hypernyms (is_a/has_a relationships) and meronyms/holonyms (part_of/ has_part relationships). Edges between concepts are given *weights* and the weight of a link is affected by the density of the HCG at that point, the depth in the HCG and the strength of connotation between the nodes. Richardson and Smeaton highlight one of the significant problems with WordNet and with hierarchical graphs – *The irregular densities of links between concepts results in unexpected semantic distance measures. These are typically as a result of expected links between concepts not being present.*

Rodríguez and Egenhofer (1999) present an approach for semantic similarity across different ontologies based on the matching process of each of the specification components in the entity class representations. The similarity function determines lexical similarities with feature sets (functions, parts, attributes). The similarity function equals the weighted sum of each specification component. The work focusses on entity classes and on comparing distinguishing features in terms of strict string matching between synonym sets that refer to those features. It is interesting to note that when undertaking human testing, the subjects' answers varied on the number of ranks used to classify entity classes. However, the authors left semantic similarity among features to future work.

Rodríguez, Egenhofer and Rugg (1999) combine feature mapping with semantic distance calculation to assess semantic similarities and provide a summary of other work that has been undertaken in comparing semantics. Their model for measuring semantic similarity has a strong linguistic basis and takes into account synonyms and different senses in the use of terms. It also considers component-object relations with properties of asymmetry in evaluation of similarity. Their work outlines a model that assesses similarity by combining feature mapping with a semantic distance measurement defined in terms of the relevance of different features in terms of the distance in a semantic network. The global similarity function is a weighted sum of the similarity values for parts, functions and attributes and yields values between 0 and 1. Context, although recognized as a relevant issue for semantic similarity, is not addressed in this work.

Sowa's Conceptual Graph Standard (Sowa 1998, 1999) provides a guide for the implementation of conceptual graphs in systems. The conceptual graph is an abstract representation for logic with nodes as concepts and conceptual relations linked together by arcs. The conceptual graphs developed within our model applies these standards and uses the operations defined within the standard as well as extending functionality with new operations to determine similarity and to specify queries.

5 Implementation

One of the major motivations of our approach is that reusable *semantic distance graphs* can be developed. These graphs could then be shared locally or made available as standardised datasets.

It is not our aim here to define precisely the nature of any SQL enhancements. Indeed, various implementations can be envisaged that do not affect the model. Rather, our aim here is to show that such enhancements can be made fairly simply and can provide a powerful and useable extension.

In our implementation[6], each of our concepts is realised through four related tables:

- A relation containing information about defined graphs, including a value for ζ for each graph,
- A relation containing defined comparators, for each concept, such as *Close_To*, *Unlike*, and so on,
- A ternary relation containing node details including τ for each node,
- A 4-ary relation containing arc details including arc distances and a flag indicating whether the arc distance is symmetric. If not then a separate tuple must be included for the reverse link.

While the graphs could be user-supplied, it would also be possible to develop libraries of commonly accepted concepts in a similar manner to that used for standardised domain-specific XML schema definitions.

As well as providing the graph searching algorithms, we extended SQL to provide support within the query language for most of the additional semantics[7]. This consisted of three parts:

1. DDL commands to define, delete and modify concept graphs.
2. Commands to define additional comparators within SQL.
3. Modifications to the syntax of the value comparator.

To support the definition of the graphs we included a `CREATE CONCEPTGRAPH` statement as follows:

```
CREATE CONCEPTGRAPH <graphname>
    FROM UNIDIRECTIONAL|BIDIRECTIONAL
        ADJACENCYTABLE <table>
        [VERTICES <table>]
        [ZOOM BY <zoomfactor>]
        [COMBINATION FUNCTION <combfunc>]
        [MAXIMUM <max_value>];
```

where

`ADJACENCYTABLE` is a ternary relation representing the domain values and the conceptual distance between them.

The optional `VERTICES` table has a binary structure and allows vertices with transition costs to be held.

The optional `ZOOM BY` clause provides for a zooming factor. If omitted it defaults to 1.

The optional `COMBINATION FUNCTION` clause provides for an alternative function for combining elements of the distance calculation. If omitted it defaults to arithmetic addition.

The optional `max_value` provides a threshold distance beyond which any distance is deemed to be the maximum. If omitted it defaults to ∞.

An attribute's definition can then be qualified as referring to a given concept graph. For example:

```
CREATE TABLE STOCKREL
    ⋮
    ITEMCOLOUR CHAR(10) CONCEPTGRAPH(COLOURS),
    ITEMTYPE CHAR(10) CONCEPTGRAPH(STOCKITEMS),
    ⋮
```

The `SELECT` statement was then extended by extending the comparator operators. This was done by defining additional operators as follows:

```
CREATE COMPARATOR <comparatorname>
    OVER <conceptgraph>
    AS <distanceconstraint>
```

where `distanceconstraint` provides a simple arithmetic function. For example,

```
CREATE COMPARATOR CLOSETO
    OVER COLOURS
    AS "< 0.3"

CREATE COMPARATOR CLOSETO
    OVER STOCKITEMS
    AS "< 0.4"

CREATE COMPARATOR UNLIKE
    OVER COLOURS
    AS "> 0.7"
```

For example, queries searching for (a) GREEN−*ish* stock items and (b) New chairs which have a (very) different colour to stock item A12, could be written:

```
SELECT ITEMNO, ITEMDESC
    FROM STOCKREL
    WHERE ITEMCOLOUR CLOSETO "GREEN"

SELECT NEWITEMNO, NEWITEMDESC
    FROM STOCKREL, NEWSTOCK
    WHERE ITEMNO = "A12"
    AND NEWITEMTYPE CLOSETO "Chair"
    AND ITEMCOLOUR UNLIKE NEWITEMCOLOUR
```

[6] For general utility, our prototype implementation is based around the relational model and the freely available `MySQL` software (MySQL n.d.).

[7] The combination function $\oplus$ was not implemented in our prototype - we used simple arithmetic addition. However, suggestions for its inclusion in SQL are provided.

Queries using concepts not explicitly defined can also be used through explicit specification in the query. For example, given an alternative colour-based distance metric of `intensity`, which might consider rich colours as closer to each other than to pale colours, we could write:

```
SELECT ITEMNO, ITEMDESC
    FROM STOCKREL
    WHERE ITEMCOLOUR
        CLOSETO(INTENSITY) "GREEN"
```

Our extensions were implemented through query interception and rewriting but could also have been implemented through direct amendment of the SQL processing.

6 Discussion and Future Directions

There are occasions when more than one distance measure is needed, either independently or in combination. Two modes of combination can be identified:

- Integration of Concepts. In this mode, concepts can be combined at a fine-grained level to allow a mixing of paths between intermediate ideas. For example, the travel time from Adelaide to Orono may be determined using many graphs, one for roads, a second one for air routes and a third for rail. In this case the closest airport might not represent the most appropriate route.
- Aggregation of Concepts. In this mode, concepts are not able to be combined. However, a measure of similarity may be ascertained through aggregating the results over many paths. The city of Adelaide in Australia, for example, could be considered similar to the city of San Diego in the USA, given both are a similar age, have similar populations, weather patterns, industrial emphases, a large island just off the coast, and so on.

One particular area that therefore suggests itself for future work is the use of combinations of concepts to find broad similarities (similar to Spanoudakis and Constantopoulos' *Aggregate Distance Metric* (Spanoudakis & Constantopoulos 1996)). For example, we might ask not only *Find all cities that are* similar *to Adelaide* where *similar* is defined over a collection of measures, but also implement a form of data mining that utilises these aggregate difference measures.

7 Summary

Determining the distance between object consistently is a common and often time-consuming problem. This paper has presented a flexible and general model for determining the semantic distance between objects, which we believe can be adopted (and understood) generally and implemented through simple graph search algorithms. Such a model can be used to promote a general agreement on the methods of applying semantic distance over a group of applications, while still allowing tailoring when required. The model facilitates the reuse of distance measures through reusable semantic distance graphs and has demonstrated the plausibility of simple enhancements to standards such as SQL.

References

Allen, J. (1983), 'Maintaining knowledge about temporal intervals', *Communications of the ACM* **26**(11), 832–843.

Fellbaum, C., ed. (1998), *WordNet: An Electronic Lexical Database*, Bradford Books.

Hornsby, K. & Egenhofer, M. (1999), Shifts in detail through temporal zooming, *in* '10th International Workshop on Database and Expert Systems Applications - Spatio-Temporal Data Models and Languages (STDML)', IEEE Computer Society, Florence, Italy, pp. 487–491.

Kainz, W., Egenhofer, M. J. & Greasley, I. (1993), 'Modelling spatial relations and operations with partially ordered sets', *International Journal of Geographical Information Systems* **7**(3), 215–229.

Kedad, Z. & Métais, E. (1999), Dealing with semantic heterogeneity during data integration, *in* J. Akoka, B. Mokrane, I. Comyn-Wattiau & E. Métais, eds, 'Eighteenth International Conference on Conceptual Modelling', Vol. 1728 of *Lecture Notes in Computer Science*, Springer, Paris, France, pp. 325–339.

Miller, R. & Yang, Y. (1997), Association rules over interval data, *in* J. Peckham, ed., 'ACM SIGMOD Conference on the Management of Data', ACM Press, Tucson, Arizona, USA, pp. 452–461.

MySQL (n.d.), 'SQL shareware software, online at `http://www.mysql.com/`'.

Richardson, R., Smeaton, A. & Murphy, J. (1994), Using wordnet as a knowledge base for measuring semantic similarity between words, Technical Report Working Paper CA-1294, School of Computer Applications, Dublin City University.

Rodríguez, M. A. & Egenhofer, M. J. (1999), Putting similarity assessment into context: Matching-distance with the user's intended operations, *in* P. Bouquet, L. Serafini, P. Brézillon, M. Benerecetti & F. Castellani, eds, '2nd International and Interdisciplinary Conference on Modeling and Using Context, CONTEXT-99', Vol. 1688 of *Lecture Notes in Artificial Intelligence*, Springer, Trento, Italy, pp. 310–323.

Rodríguez, M., Egenhofer, M. & Rugg, R. (1999), Assessing semantic similarities among geospatial feature class definitions, *in* A. Vckovski, K. Brassel & H.-J. Schek, eds, 'Second International Conference on Interoperating Geographic Information Systems, INTEROP'99', Vol. 1580 of *Lecture Notes in Computer Science*, Springer, Zurich, Switzerland, pp. 189–202.

Sowa, J. (1999), 'Conceptual graph standard', online at `http://www.bestweb.net/~sowa/cg/cgstand.htm`.

Sowa, J. F. (1998), Conceptual graph standard and extension, *in* M.-L. Mugnier & M. Chein, eds, '6th International Conference on Conceptual Structures, ICCS '98', Vol. 1453 of *Lecture Notes in Computer Science*, Springer, Montpellier, France, pp. 3–14.

Spanoudakis, G. & Constantopoulos, P. (1994), Similarity for analogical software reuse: A computational model, *in* '11th European Conference on Artificial Intelligence(ECAI '94)', Amsterdam, The Netherlands, pp. 18–22.

Spanoudakis, G. & Constantopoulos, P. (1996), 'Elaborating analogies from conceptual models', *International Journal of Intelligent Systems* **11**(11), 917–974.

Stevens, S. (1946), ‘On the theory of scales of measurement’, *Science* **103**(2684), 677–680.

Weinstein, P. & Birmingham, W. (1999), Agent communication with differentiated ontologies: eight new measures of description compatibility, Technical report, Department of Electrical Engineering and Computer Science, University of Michigan.