

# Using a Similarity Measurement to Partition a Vocabulary of Medical Concepts

Huanying (Helen) Gu<sup>1</sup>, James Geller<sup>1</sup>, Li-min Liu<sup>1</sup>, and Michael Halper<sup>2</sup>

<sup>1</sup> CIS Dept., New Jersey Institute of Technology, Newark, NJ 07102 USA  
{helen, geller, limin}@homer.njit.edu

<sup>2</sup> Math & Computer Science Dept., Kean University, Union, NJ 07083 USA  
mhalper@turbo.kean.edu

**Abstract.** Controlled medical vocabularies have become increasingly important in a range of medical informatics applications. However, the extensive size of most vocabularies often makes it difficult for users to gain an understanding of their contents. In previous work, we have investigated the partitioning of a large semantic-network based medical vocabulary into smaller units, for the purpose of easier graphical display and comprehension. The partitioning process relied heavily on a domain expert. In this paper, we propose a structural method for automating the partitioning of a vocabulary. The structural method is based on a definition of the similarity of a pair consisting of a child concept and its parent concept in the semantic network. A distribution over these similarities for all pairs in the semantic network is then computed. Based on this distribution, the semantic network can be partitioned into more manageable pieces. The approach has been applied to the InterMED and a complex portion of the MED, two large medical vocabularies.

## 1 Introduction

In order to improve the quality of medical communications, the medical community has created a number of standardized vocabularies, e.g., [10, 15, 22, 29]. These vocabularies help healthcare providers, insurance companies, pharmacies, and other members of the medical community avoid misunderstandings and define unique encodings for drugs, diagnoses, diseases, procedures, etc. Some of these vocabularies have been computerized, such as the MED [8, 9] and the InterMED [23, 26] which are modeled as semantic networks [13, 28].

The MED is a semantic network with over 48,000 concepts, 61,000 IS-A links and 71,000 other links in its 1996 version. While the MED is extremely useful, it is difficult for its users to grasp the knowledge contained in it. Intuitively, a graphical representation of the MED should help. However, one can estimate that a picture of the MED would comprise an area of at least 300 square feet, which is by far too large for any comprehension purposes. For the InterMED, we measured the size of a display. It required over sixty square feet for about 3,000 concepts and all the IS-A links connecting them.

In order to deal with this problem, we have investigated the approach of partitioning a semantic network into smaller disjoint units, called “contexts,”

that fulfill three conditions: (1) Every context should be meaningful in the eyes of an expert; (2) Every context should fit onto one screen; and (3) The union of all disjoint contexts should be identical to the original network.

Semantic networks are directed acyclic graphs (DAGS) that are constructed around backbones of IS-A links. One would think that the extensive literature on graph partitioning [1, 2, 3, 17] would supply the theoretical means for achieving the kind of partitioning described above. Unfortunately, this expectation fails for two reasons: (1) Partitioning problems tend to be NP-complete [12], i.e., probably not solvable by polynomial algorithms; (2) The requirement of “meaningful groups” is beyond the scope of graph algorithms.

To overcome these problems, we have previously combined human expert judgment with algorithmic tools. We have introduced a technique called “disciplined modeling” [14, 24] that results in a partitioning of a semantic network into a set of trees. However, in this approach, human expert judgment is an important ingredient. In this paper, we present an approach which completely avoids the involvement of a human expert. Rather, it relies on structural features of the semantic network.

To understand the basic idea of our partitioning approach, we need some background regarding the MED and InterMED. We will only refer to the InterMED, but all ideas apply equally to the MED. In the InterMED, each attribute and relationship is introduced at a unique concept and is inherited by all concepts below it, via the IS-A links.

Attributes describe information local to a concept, while relationships are links between concepts. In Fig. 1, we show a small subnetwork of the InterMED. Round-edged rectangles are concepts, and attributes are listed beneath the concept names. Bold arrows stand for IS-A links. Labeled arrows are relationships. The fact that a property is introduced at a certain point, however, does not guarantee that a value is also introduced for it at that point. Attribute values are *not* inherited in the InterMED, but relationship targets usually are. In some cases, relationships are overridden at lower levels in the hierarchy.

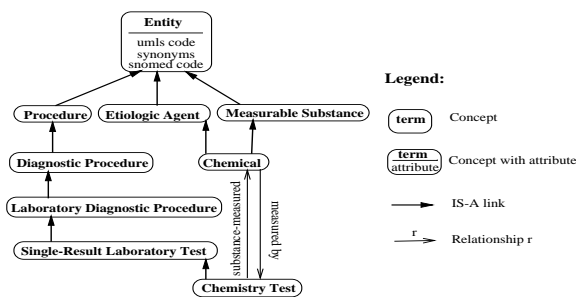


Fig. 1. A subnetwork of the InterMED

Sometimes concepts in the InterMED are very similar to their parent con-

cepts, while at other times they are very different. A child might not introduce any new attributes and relationships. In addition, it might inherit all relationship targets without any overriding. Clearly, such a child concept is very similar to its parent concept. On the other hand, a child might introduce several new attributes and relationships and override the targets of all its inherited relationships. In this case, the child concept is very different from its parent.

The underlying idea of the partitioning is to examine every concept in the InterMED and find out how similar it is to its parent(s). Obviously, if a concept  $N$  is very different from (all) its parent(s), we may suspect that a new meaningful group of concepts starts just below it.

The similarity measure we propose can be expressed concisely by a single number. By using either the introduction points of properties alone, or by combining them with target inheritance behavior, we gain some fine control over the computation of similarity numbers. In this approach, no human expert is required at all.

The rest of this paper is organized as follows. In Section 2, we will describe the approach that we have used to partition the InterMED. Section 3 compares our results with the results obtained by our previous approach, which made use of a human expert. In Section 4, we will briefly review some related literature. Finally, in Section 5, we will conclude with a discussion of the value of these results and future work.

## 2 Partitioning Approach

Our partitioning approach is based on the similarities of the property sets of child/parent pairs in the semantic network. The property similarity of a child/parent pair gives a quantitative measure of the similarity of the children and parents. It is obtained from comparisons of the respective properties of each. We define the property similarity  $\sigma$  of a child/parent pair as a number between 0 and 1, where 0 means the lowest and 1 the highest similarity (identical).

### 2.1 Similarity based on Property Introduction

In the InterMED, each property is first introduced at a unique concept which we will call *property-introducing concept* for the property [19]. A concept may serve as the property-introducing concept for many properties. A property is inherited by all the children and descendants of a concept. Thus, there are only two cases for all child/parent pairs. One is that the child concept has the same properties as its parent concept. This means that the child concept only inherits the properties from its parent concept instead of introducing any new properties of its own. This child concept is obviously similar to its parent concept. For this case, we assign this pair the similarity 1. The other case is that the child concept has more properties than its parent. Either the child concept introduces at least one new property, or it inherits properties from multiple parents with different properties. In this case, the two concepts will be given the similarity 0.

We can partition the network into several subnetworks by removing all child/parent links which have similarities 0. Thus, each subnetwork will contain concepts with the same properties. One could argue for a more sophisticated numeric evaluation. For instance, we could use the following formula.

$$\sigma = \frac{\text{number of properties at parent}}{\text{number of properties at child}} \quad (1)$$

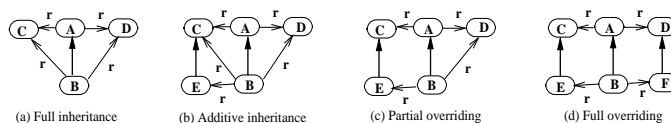
However, the InterMED contains only 58 distinct properties and there are only 38 child/parent pairs in which the child concepts have more properties than their parents. Thus, we obviously want to cut all child/parent links where the child and parent are at all different. Therefore, assigning 0 to such pairs of concepts is a good choice. In a network that has a higher degree of property introduction, we would use Formula 1. After we remove all links with similarities 0, the InterMED with 2,820 concepts is divided into 38 subnetworks. Unfortunately, one of the subnetworks contains more than 1,000 concepts. Thus, we need a better formula to compute the property similarity.

## 2.2 Similarity based on Relationship Overriding

As we saw, considering only the numbers of properties for computing the similarity of each child/parent pair does not give a satisfactory partitioning result. The property similarities need to be calculated more accurately.

Before giving the formulas which we used to compute the property similarity, we need to recall that attributes have literal values and relationships point to other concepts. Both attributes and relationships may be set-valued. That means that one relationship may have several target concepts. The relationship inheritance in the InterMED may take place in four different ways:

- *Full inheritance*: A child concept inherits all values (target concepts) of the relationship. In Fig. 2 (a), *A* has the relationship targets *C* and *D*. *B* inherits the relationship *r* and, therefore, has the same targets *C* and *D*.



**Fig. 2.** Four different ways of relationship inheritance

- *Additive inheritance*: A child concept inherits all values of the relationship and adds more values for the relationship. In Fig. 2 (b), *A* has the relationship targets *C* and *D*. *B* inherits the relationship *r* and has the same targets *C* and *D*. In addition, *B* has the target *E*, which is a child of *C*.

- *Partial overriding*: A child concept inherits some values and overrides (refines) other values of the relationship. In Fig. 2 (c),  $B$  inherits the target  $D$  from  $A$ . However, the target  $C$  is overridden by the new target  $E$ , which is a child of  $C$ .
- *Full overriding*: A child concept overrides (refines) all values of the relationship. In Fig. 2 (d), all targets of the relationship  $r$  of  $A$  are overridden.  $B$  refers to  $E$  and  $F$  instead of their respective parents,  $C$  and  $D$ .

We will now introduce a similarity formula that takes the possibilities listed above into account. Consider a child having  $m$  relationships and its parent having  $n$  relationships. The number of fully overridden relationships is  $f$ , the number of partially overridden relationships is  $p$ , and the number of additive inheritance relationships is  $d$ . Then the similarity of the child/parent pair is:

$$\sigma = \frac{n - f - p - d}{m} \quad (2)$$

When the child has the same number of relationships as its parent ( $m = n$ ) and no relationships are overridden, the similarity of the child/parent pair will be 1. On the other hand, if there is no full inheritance for any relationship, the similarity for the child/parent pair will be 0.

Using Formula 2, we can compute the property similarities for all child/parent pairs. Then, we can remove IS-A links between child concepts and their parents for pairs with low similarities. Ideally, the network will be partitioned into several smaller sub-networks, within each of which all concepts are quite similar.

Unfortunately, due to multiple inheritance, this is not always true. Because the InterMED is a DAG, cutting “random edges” gives us little control over partitioning into components. For example, if after cutting, each child is still connected to at least one parent, the original network will not be partitioned at all. If we can first reduce the graph to a tree, then removing links from the tree will result in a forest. There are additional advantages to working with a tree. It is generally easier to comprehend a tree than a DAG consisting of the same concepts, because in a tree upward paths are not branching. Therefore, we reduce the DAG to a tree as follows. We call this step *tree identification*.

Assume that a child concept  $C$  has  $q$  parent concepts ( $q > 1$ ). The property similarities of child/parent pairs are  $\sigma_1, \sigma_2, \dots, \sigma_q$  respectively. If there is only one maximum number, call it  $\sigma_{max}$ , among  $\sigma_1, \sigma_2, \dots, \sigma_q$ , then all links with similarities  $\sigma_i$  ( $i \neq max$ ) will be removed. If there are two or more maximum numbers among  $\sigma_1, \sigma_2, \dots, \sigma_q$ , one IS-A link with maximum similarity  $\sigma_{max}$  will be retained randomly, and all others will be removed.

Because *tree identification* will produce a tree, removing any links in the tree will result in a forest of more than one tree. Now the question is: Which links of the tree should be removed to create a useful partitioning result? Because the purpose of partitioning is to help users comprehend the concept network, each subtree produced by the partitioning scheme should be meaningful and have a manageable size. After we compute the similarities for all child/parent pairs, the distribution of property similarities can be calculated (Table 1). According to

Table 1, there are  $n_1$  child/parent pairs that have property similarities between 0 and  $k_1$ , etc. Table 1 gives us the similarity distribution for the whole network and helps us decide which concepts should reside in the same context. We can choose a numeric parameter  $K$  and remove all the IS-A connections between child concepts and their parent concepts for which the similarity  $\sigma < K$ . The result will be several trees. All child/parent pairs in each tree have similarities greater than  $K$ . By varying  $K$ , we have some control over the number of links that are cut. With that, we get some control over the size of the contexts that are generated.

**Table 1.** Distribution of similarities

Property Similarities	Number of child/parent pairs
0	$n_0$
(0, $k_1$ )	$n_1$
[ $k_1$ , $k_2$ )	$n_2$
$\vdots$	$\vdots$
[ $k_i$ , 1)	$n_{i+1}$
1	$n_{i+2}$

$$0 < k_1 < k_2 < \dots < k_i < 1$$

**Table 2.** Similarities of the InterMED

Property Similarities	Number of child/parent pairs
0	855
(0, 0.1)	0
[0.1, 0.2)	0
[0.2, 0.3)	1
[0.3, 0.4)	17
[0.4, 0.5)	0
[0.5, 0.6)	72
[0.6, 0.7)	775
[0.7, 0.8)	22
[0.8, 0.9)	212
[0.9, 1.0)	1
1	864

**Table 3.** Similarities of complex network MED

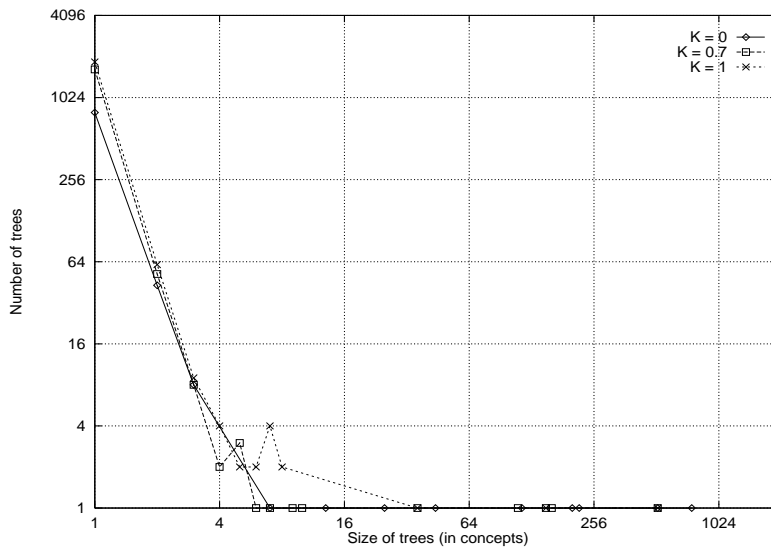
Property Similarities	Number of child/parent pairs
0	0
( 0, 0.1 )	0
[0.1, 0.2)	0
[0.2, 0.3)	0
[0.3, 0.4)	0
[0.4, 0.5)	1
[0.5, 0.6)	2
[0.6, 0.7)	4
[0.7, 0.8)	2
[0.8, 0.9)	3
[0.9, 1.0)	0
1	26

An interesting question is whether we gain anything if we combine similarity by property introduction with similarity by relationship overriding. The answer is “no,” because the contribution of relationship introduction is already contained in Formula 2. For example, if a child has more relationships than its parent due to relationship introduction, the link will not be assigned a similarity of 1, even if the child has full inheritance for all relationships. The contribution of attribute introduction is negligible because there are only 12 attributes in a vocabulary of 2,820 concepts, most of which are introduced at the root.

### 2.3 Partitioning the InterMED

Now let us use the approach described above to partition the InterMED. The InterMED contains 2,820 medical concepts. The number of child/parent pairs is 4,687. After *tree identification*, the distribution of property similarities can be computed (Table 2). Based on this distribution figure, we can choose a  $K$  to partition the InterMED by removing all links with similarities less than  $K$ . We can vary  $K$  to obtain alternative partitioning results.

First, let us choose  $K = 1.0$ . This means that each subtree resulting from the partitioning will contain concept pairs with similarities 1. The number of subtrees created by this partitioning is 1,955; the biggest subtree contains 519 concepts; there are 1,868 subtrees with only one concept. Fig. 3 shows the number of trees for each size of the trees.



**Fig. 3.** Size of trees vs. number of trees for different values of  $K$

For  $K = 0.7$ , the partitioning gives us 1,720 subtrees, containing the child/parent pairs with similarities greater than 0.7 ( Fig. 3). The biggest tree consists of 519 concepts. The number of trees consisting of a single concept is 1,646. Compared with  $K = 1.0$ , this result is better because even though the biggest tree is still of the same size, the number of small trees is reduced. However, there are still too many trees with small numbers of concepts. With  $K = 0$ , the partitioning will produce a forest, with trees containing child/parent pairs with similarities greater than 0. The number of trees is 855. The largest contains 758 concepts. There are 795 trees which consist of only one concept (Fig. 3).

As we have seen, different values of  $K$  can be chosen to partition the InterMED differently. Unfortunately, we do not get ideal partitioning results. Some of trees contain large numbers of concepts, which cannot be displayed neatly on one screen. There are also many trees consisting of only a single concept or very few concepts; such trees do not capture much meaning.

The appearance of a large number of single-concept trees or trees with very few concepts is due to the incompleteness of the the InterMED; among the 2,820 concepts of the InterMED, 2,186 are leaves. Most of the single-concept trees are derived from the leaves of the original network. If more concepts were added to

the InterMED, some of these leaves would become parents and the trees would turn into contexts with significant numbers of concepts. The remaining large trees would need to be partitioned by a human. Our partitioning algorithm still improves the situation, because there are few large trees left. In addition, even the largest of those trees is considerably smaller than the original vocabulary.

### 3 Structural Partitioning vs. Expert Partitioning

In this section, we will compare our partitioning results with the results obtained by semantic partitioning [14]. We apply these two methods to the most complex subnetwork of the MED. The results turn out to be quite similar.

In the MED, the concept **Cortisporin Ophthalmic Ointment** has the most ancestors: 39. We will focus on the subnetwork containing this concept and all its ancestors. The subnetwork contains 62 IS-A relationships and 157 other relationships. For 62 child/parent pairs, we use Formula 2 to compute their similarities. After applying *tree identification*, we compute the similarity distribution of the tree ( Table 3). If we want only child/parent pairs with maximum similarity to reside in the same tree, we can choose  $K = 1$ . After removing all IS-A links with  $\sigma < 1$ , we obtain a forest with 13 trees (Fig. 4).

In [14], we described a methodology to partition a network into several trees. There, a domain expert is required to make a judgment about whether a child concept and its parent concept are similar. Using that approach, the complex subnetwork was partitioned into 18 trees (Fig. 5).

Comparing the results obtained from the two approaches (Fig. 4 and Fig. 5), we find that our “structural” approach gives results that are similar to the results of the semantic approach in [14]. The results of the structural approach also appear semantically plausible. There are 10 tree roots out of 13 that are the same as for the semantic partitioning obtained from a domain expert’s knowledge.

### 4 Related Literature

The issue of grouping concepts together in a “reasonable” manner has long been known as “conceptual clustering” in AI. “Clustering is usually viewed as a process of grouping physical or abstract objects into classes of similar objects. One needs to define a measure of similarity between the objects and then apply it to determine classes” [21]. A “goodness measure” is usually defined for the overall partitioning of objects [7]. Note that these are not classes in the sense of object-oriented programming, but classes in the sense of conceptual categories. On the other hand, in statistical clustering [11] and numerical taxonomy [27], most similarities are defined between pairs of objects. Our formula considers child/parent pair similarity and is applied to all concepts in order to partition an entire network into a collection of trees.

In [16], we find one of the oldest AI approaches to this problem, which partitions networks into “net spaces.” These net spaces delimit the scopes of quantified variables. The partitioning into net spaces is done by experts and it cannot

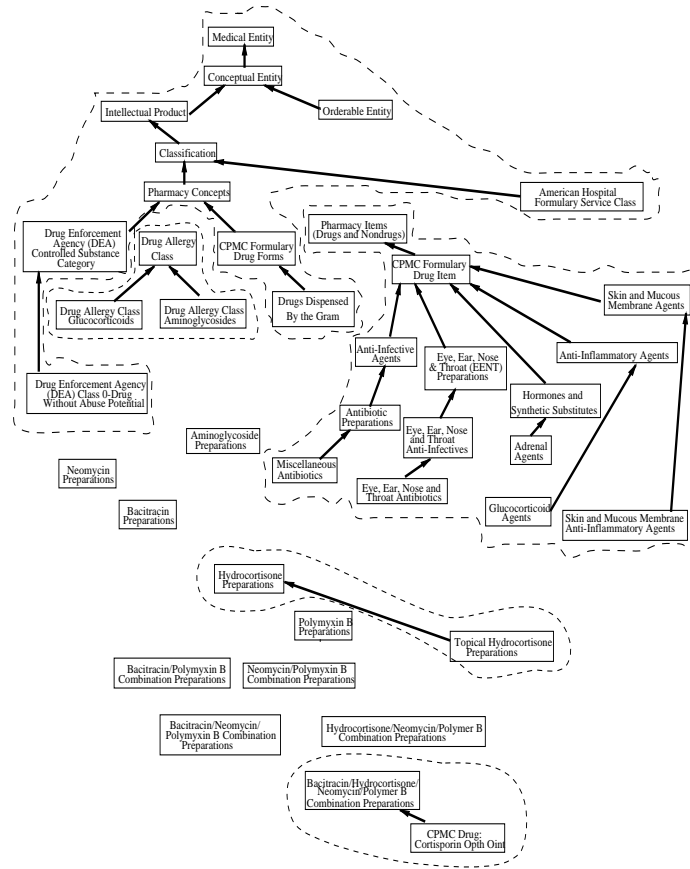


Fig. 4. Partitioning result based on structural partitioning

be carried out by checking the similarity among concepts. Different from our vocabulary networks, in SNePS [25] the IS-A relation is treated in the same way as all other relations. In [18], Levinson presents the principle of pattern-associativity by indexing objects into multi-levels. This approach allows one to organize conceptual graphs into a multi-level partial order by subgraph-isomorphism.

In [30], Woods describes the taxonomies of structured conceptual descriptions following work of the KL-ONE family [5]. Such taxonomies were generated by the subsumption relationship that relates each pair of concepts [13]. Our formula provides an approach to break the IS-A hierarchy in a reasonable way to generate a collection of trees.

To answer queries using different knowledge representations at different abstraction levels, Chu [6, 20] proposed the “Type Abstraction Hierarchy” which characterizes the instance values differently at different knowledge levels. However, the instances have neither subsumption relationships nor attributes among



We have obtained an important result, namely, that the outcome of a human expert-based partitioning of a large vocabulary is quite similar to the partitioning which is the result of applying our purely structural method.

In future work, we will further refine our similarity measures by in turn computing the similarity of every single partially overridden relationship. Another interesting idea is to combine the structural and expert-based approaches. The task of partitioning a vocabulary of thousands or tens of thousands of concepts into manageable groups can be overwhelming and by far too time consuming. As an expert is needed for the semantic partitioning, this can be very expensive! In short, the semantic approach by itself is not realistic for large vocabularies, and large vocabularies are the only interesting ones. We can use our structural approach to help a domain expert. The results of the structural partitioning method can be given to a domain expert who can then work on partitioning the remaining large trees into smaller subtrees.

## References

1. Agasi, E., Becker, R. I., Perl, Y.: A shifting algorithm for constrained min-max partition on trees. *Discrete Applied Mathematics* **45** (1993) 1–28
2. Becker, R. I., Perl, Y.: The shifting algorithm technique for the partitioning of trees. *Discrete Applied Mathematics* **62** (1995) 15–34
3. Becker, R. I., Perl, Y., Schach, S.: A shifting algorithm for min-max tree-partitioning. *J. ACM* **29** (1982) 56–67
4. Bertino, E., Martino, L.: *Object-Oriented Database Systems: Concepts and Architectures*. Addison-Wesley Publishing Company, New York (1993)
5. Brachman, R. J., Schmolze, J.: An overview of the KL-ONE knowledge representation system. *Cognitive Science* **9** (1985) 171–216
6. Chu, W. W., Chen, Q., Lee, R.: Cooperative query answering via type abstraction hierarchy. In: *Proc. Int'l Working Conference on Cooperating Knowledge Based Systems*. University of Keele, UK (1990) 271–290
7. Chu, W. W., Chiang, K.: Abstraction of high level concepts from numerical values in databases. In: *Proc. AAAI Workshop on Knowledge Discovery in Databases*. Seattle, WA (1994) 133–144
8. Cimino, J. J., Barnett, G. O.: Automated translation between medical terminologies using semantic definitions. *MD Comput.* **7** (1990) 104–109
9. Cimino, J. J., Clayton, P. D., Hripcsak, G., Johnson, S.: Knowledge-based approaches to the maintenance of a large controlled medical terminology. *JAMIA* **1** (1994) 35–50
10. *College of American Pathologists: The Systematized Nomenclature of Medicine*. Skokie, IL (1982)
11. Everitt, B.: *Cluster Analysis*. Heinemann Educational Books, London (1980)
12. Gary, M. R., Johnson, D. S.: *Computers and Intractability*. Freeman, New York (1979)
13. Gregor, R. M.: The evolving technology of classification-based knowledge representation systems. In: Lehman, F. (ed.): *Semantic Networks in Artificial Intelligence*. Pergamon Press, Oxford, UK (1992) 385–400
14. Gu, H., Perl, Y., Geller, J., Halper, M., Singh, M.: A methodology for partitioning a vocabulary hierarchy into trees. *Artificial Intelligence in Medicine* **15** (1999) 77–98

15. Health Care Financing Administration: International Classification of Diseases: 9th Revision, Clinical Modification: ICD-9-CM. 4th edn. Washington, DC (1991)
16. Hendrix, G. G.: Encoding knowledge in partitioned networks. In: Findler, N. V. (ed.): *Associative Networks: Representation and Use of Knowledge by Computers*. Academic Press, Inc., New York (1979) 51–92
17. Kundu, S., Misra, J.: A linear tree-partitioning algorithm. *SIAM J. Comput.* **6** (1977) 131–134
18. Levinson, R.: Pattern associativity and the retrieval of semantic networks. In: Lehman, F. (ed.): *Semantic Networks in Artificial Intelligence*. Pergamon Press, Oxford, UK (1992) 573–600
19. Liu, L., Halper, M., Geller, J., Perl, Y.: Controlled vocabularies in OODBs: Modeling issues and implementation. *Distributed and Parallel Databases* **7** (1999) 37–65
20. Merzbacher, M., Chu, W. W.: Pattern-based clustering for database attribute values. In: *Proc. AAAI Workshop on Knowledge Discovery in Databases*. Washington, DC (1993) 291–298
21. Michalski, R. S., Stepp, R. E.: Clustering. In Shapiro, S. C. (ed.): *The Encyclopedia of Artificial Intelligence*. 2nd edn. John Wiley & Sons, New York (1992)
22. National Library of Medicine: *Medical Subject Headings*. Bethesda, MD (1997; updated annually)
23. Oliver, D., Shortliffe, E.: Collaborative model development for vocabulary and guidelines. In: Cimino, J. J. (ed.): *Proc. '96 AMIA Annual Fall Symposium*. Washington, DC (1996) 826
24. Perl, Y., Geller, J., Gu, H.: Identifying a forest hierarchy in an OODB specialization hierarchy satisfying disciplined modeling. In: *Proc. CoopIS'96*. Brussels, Belgium (1996) 182–195
25. Shapiro, S. C., Rapaport, W. J.: The SNePS family. In: Lehman, F. (ed.): *Semantic Networks in Artificial Intelligence*. Pergamon Press, Oxford, UK (1992) 243–275
26. Shortliffe, E., Barnett, G., Cimino, J. J., Greenes, R., Huff, S., Patel, V.: Collaborative medical informatics research using the Internet and the World Wide Web. In: *Proc. '96 AMIA Annual Fall Symposium*. Washington, DC (1996) 125–129
27. Sneath, P. H. A., Sokal, R. R.: *Numerical Taxonomy: The Principles and Practice of Numerical Classification*. W. H. Freeman and Co., San Francisco, CA (1973)
28. Sowa, J. F. (ed.): *Principles of Semantic Networks*. Morgan Kaufmann Publishers, Inc., San Mateo, CA (1991)
29. US Dept. of Health and Human Services, NIH, National Library of Medicine: *Unified Medical Language System* (1998)
30. Woods, W. A.: Understanding subsumption and taxonomy: A framework for progress. In Sowa, J. F. (ed.): *Principles of Semantic Networks*. Morgan Kaufmann Publishers, Inc., San Mateo, CA (1991) 45–94