Semantic refinement and error correction in large terminological knowledge bases ☆

James Geller [a,*], Huanying Gu [b], Yehoshua Perl [a], Michael Halper [c]

[a] *CS Department, New Jersey Institute of Technology, 323 Dr. King Blvd., Newark, NJ 07102, USA*
[b] *Department of Health Informatics, University of Medicine and Dentistry, Newark, NJ 07107, USA*
[c] *Mathematics & Computer Science Department, Kean University, Union, NJ 07083, USA*

Abstract

Capturing the semantics of concepts in a terminology has been an important problem in AI. A two-level approach has been proposed where concepts are classified into high-level semantic types, with these types constituting a portion of the concepts' semantics. We present an algorithmic methodology for refining such two-level terminologic networks. A new network is produced consisting of ''pure'' semantic types and intersection types. Concepts are uniquely re-assigned to these new types. Overall, these types form a better conceptual abstraction, with each exhibiting uniform semantics. Using them, it becomes easier to detect classification errors. The methodology is applied to the UMLS.
© 2002 Elsevier Science B.V. All rights reserved.

Keywords: Concept hierarchy; Semantic type; Semantic refinement; Terminological knowledge base; Semantic error correction

1. Introduction

Defining the semantics of natural language is one of the holy grails of Artificial Intelligence. When semantic networks were initially conceived of, their intention was to capture the semantics of natural language terms and concepts. Indeed, Quillian's father of all semantic networks [30]

* Corresponding author. Fax: +1-973-596-5777.
E-mail address: geller@oak.njit.edu (J. Geller).

was really a dictionary in which terms had been crosslinked. As semantic networks continued to be developed, opinions split on what the actual semantics of a concept is. On one hand was the opinion that the semantics of a concept is defined by "everything the concept is connected to" [30]. On the other hand was the more radical opinion, expressed by McDermott [26], that semantic networks are not semantic at all.

Unfortunately, it is impossible to precisely express the semantics of natural language by any metalanguage because this leads to an infinite regression about how to specify the semantics of the metalanguage. Having realized this fact, many researchers started to pursue other issues in semantics. One problem is exactly how to represent the meaning of complicated English sentences involving difficulties such as nested quantifier disambiguation [28]. Such complicated sentences are in some approaches represented as (propositional) semantic networks [32,33]. The meanings of the elementary terms that are composed into a larger logical structure are typically not addressed. Some attempts have been made to capture term meanings by lexical semantics [18].

In contrast to propositional semantic networks, research on inheritance networks of the KL-ONE family [42] turned away from the problems of semantics towards a number of other issues, such as the trade-off between expressive power and computational complexity in an inheritance network [5], and the relationship between networks and logics, which resulted in the development of description logics [4]. To deal with computational efficiency problems, description logics have been combined with special techniques such as parallel processing [3].

Inheritance networks have a hierarchical (tree-based) or directed acyclic graph (DAG) structure. In a series of papers, Wille [40,41] has developed an alternative method of knowledge representation based on lattices. In his Formal Concept Analysis, concepts and attributes are integrated into a single lattice structure. Recently, there has been great interest in applications of Formal Concept Analysis [17] and in the integration of Formal Concept Analysis with other formalisms of knowledge representation, such as Conceptual Graphs [33,34].

Some researchers completely abandoned tree or graph-based semantic models and turned to axiomatic semantics. However, axiomatic semantics has been used primarily in programming languages [8]. Other researchers are satisfied with the Tarski-based [36] assignment of symbols to objects, function symbols to functions, etc. Thus, they rely on the existence of an abstract mapping to real-world entities. The problem here is that another kind of regression is introduced, as it is often not possible to define the real-world entities with the necessary precision.

In the semantic network paradigm, the idea that the "meaning of a concept is defined by everything it is connected to" also leads to an infinite (and often circular) regression. Some researchers have tried to address this problem by symbol grounding [13]. Certain concepts have a meaning that is defined immediately by grounding them to perceptual inputs or discernible mental states ("pain"), and other concepts derive their meaning from those concepts.

Our approach to the semantics of a semantic network is that it is impossible to precisely define the meaning of most concepts, except in well defined, abstract areas such as mathematics. However, we accept the claim that the meaning of one concept is constrained by the other concepts it is connected to. Thus, if a concept A has a parent B to which it is connected by an IS-A link, then the meaning of A is constrained to be more specific than the meaning of B. Similarly, if a concept A has a part-of relationship to a concept C, then A is constrained to be a part of C, which typically means that A is smaller than C and often implies that A is physically connected to other parts of C. In other words, we are not giving definitive expressions capturing the full semantics of

such concepts, but we can express that all concepts that have, e.g., IS-A relationships to B and part-of relationships to C have a great deal in common. Thus, for us the question is whether it is possible to improve the mechanisms of semantic networks to further constrain the meanings of concepts to smaller sets of concepts with useful similarities.

In this context, it is instructive to look at WordNet [27], which is the best-known large, hierarchically organized source of word knowledge available on the Web. WordNet provides very little semantics. As Barker et al. write [1], WordNet

> provide[s] very shallow semantics… For each English word, these ontologies give its senses along with their definitions, parts of speech, subclasses, superclasses and sibling classes. The definitions are free text (of limited use to computer programs) and the encoded relations are the only semantics.

To this, one has to add the fact that WordNet organizes terms into Synset (Sets of Synonyms). While it is impossible to specify the exact meaning of any one Synset, all members of a Synset are constrained in their semantics: their semantics is identical. We extend this minimal idea of semantics by specifying sets of terms with semantic similarities that go beyond synonymy.

In this paper, we have developed a technique that allows us to create, based on a given classification, relatively small sets of terms with common (but not synonymous) meanings. [1] Our methodology is based on the following observations: (1) When dividing a knowledge base into microtheories, as it is done in CYC [19], it is extremely difficult to keep different parts of the knowledge base semantically consistent with each other. (2) Even within the microtheory of a single expert, there are likely to be inconsistencies, because human categories and biases tend to shift over time [2].

Thus, there are fundamental differences between a small semantic network designed by one knowledge engineer in a short period of time and a large semantic network designed over a long time by several knowledge engineers. In a small, well designed semantic network one can rely on the fact that every concept correctly generalizes all its descendants. In other words, 'every concept is an abstraction of all its descendants'. This is not always true in a large semantic network where certain important concepts will be likely to occur simultaneously in the portions designed by different knowledge engineers. This is a major deficiency which must be overcome by integration. However, integrating the complete network from the individual pieces is very difficult. Similarly, designing a consistent semantic network by intensive communication between all designers is close to impossible, too. Furthermore, in some cases the task of semantic modeling suffers from more than shifting biases and individual perspectives of a knowledge engineer. Due to its nature of requiring high degrees of specialization and an excellent human memory, knowledge modeling is often error prone.

In summary, it is difficult to design large semantic networks that have the property that every concept is truly an abstraction of everything under it. Thus, it is desirable to introduce some abstraction that recaptures this lost property. For this purpose, we introduce a new network of high-level abstract concepts which we call *semantic types*. From now on, we will use the name "semantic network" *only* for this (inheritance) network of semantic types. Now we need a new

---

[1] The precise meaning of "relatively small" will become clear later on.

name for the pre-existing semantic network. We will call the preexisting semantic network the *thesaurus.*

Next, we need to assign every concept in the thesaurus to at least one semantic type. The resulting structure, consisting of (1) *thesaurus*, (2) *semantic network, and* (3) *assignments of concepts from the thesaurus to semantic types in the semantic network*, recaptures the desirable property that certain high-level abstractions are assigned to all concepts that this abstraction generalizes.

We will refer to a structure that consists of a thesaurus, a semantic network, and appropriate assignments as a *Terminological Knowledge Base.* At first, it might appear that this new representation format does not gain any additional semantic insights. However, as we will show in this paper, it becomes possible to constrain the semantics of concepts in the thesaurus considerably, by generating smaller groups of concepts which are uniform in the sense that they are assigned to a unique semantic type or to a unique combination of semantic types.

1.1. An outline of our semantic refinement methodology

We will now summarize our semantic refinement methodology, which consists of a series of six steps. The details will be supplied in the balance of this paper.

In **Step 1**, the concepts of a domain are collected, in the same way as one operates when building a more traditional knowledge base. This collection step can be performed by several experts with very little interaction. As mentioned earlier, we will refer to the collection of concepts obtained as the thesaurus. The significance of this name will become clear later on.

Then, in **Step 2**, a small set of *original semantics types* is defined. These are supposed to classify all the collected concepts. In other words, every one of the concepts belongs to at least one original semantic type. One may think of these original semantic types as higher level concepts. IS-A relationships and other semantic relationships are also defined between these original semantic types. Together, the original semantic types and the semantic relationships form what is called (for historical and practical reasons) the *semantic network*. It is indeed a fairly traditional semantic network [42], because the IS-A relationships form the backbone of this knowledge structure.

The process of creating the semantic network requires the collaboration of knowledge engineers and domain experts and is the only part of the methodology of this paper which demands the cooperation of a larger group of experts. As the semantic network is typically several orders of magnitude smaller than the thesaurus, this demand is deemed acceptable.

In **Step 3**, human experts assign concepts (presumably those that they originally contributed to the thesaurus) to the most specific original semantic types. Every concept needs to be assigned to at least one original semantic type. However, if a concept is assigned by one researcher to several original semantic types, that is acceptable. Similarly, if a concept happens to occur in two or more "microtheories" of the thesaurus, each occurrence may be assigned independently (i.e., by several domain experts) to several original semantic types. If different experts make differing assignments, that is not considered a problem in our methodology.

Step 4 defines the core of our methodology. First, a set of additional semantic types is generated algorithmically. We call these additional semantic types *intersection types*. Secondly, the original semantic types are transformed into *pure semantic types*. The pure semantic types are identical to the original semantic types both in name and semantic network location; however, the sets of concepts

assigned to them are different. The term *new semantic types* refers to both pure semantic types and intersection types. It stands in contrast to the term original semantic types. In symbols, we can say:

new semantic types = pure semantic types ∪ intersection types.

The method of generating intersection types will be explained below in detail. However, the basic idea is that any unique combination of original semantic types defines an intersection type if and only if there is at least one concept belonging exactly to all original semantic types of this combination. As a second result of this step, every concept is assigned to one unique new semantic type. Thus, this classification of concepts by new semantic types is a *disjoint classification*.

Step 5 of our methodology is also an algorithmic step. IS-A links are inserted between pairs of intersection types or between intersection types and pure semantic types. As a result, the original semantic network is extended and transformed into the *augmented semantic network* (briefly: augmented network).

In **Step 6**, domain experts review the results of the two previous algorithmic steps. Our experience shows that they will detect modeling deficiencies, i.e., true errors, as opposed to different semantic perspectives. These kinds of errors include omissions, non-uniform classifications, classification errors, ambiguities, redundant classifications, and independently listed synonyms. Every one of these problems will be discussed in Section 4 in more detail.

The reasons why our methodology makes it easier to detect errors will now be summarized. As should be clear by now, our approach does not lead to a complete semantic classification of concepts. Rather, it takes as input fairly large sets of concepts that are classified by original semantic types and returns as output smaller (and sometimes much smaller) disjoint sets of concepts, where each concept is *uniquely* classified either by a pure semantic type or by an intersection type. If an intersection type has a very small number of concepts assigned to it, it is likely that a classification error has emerged. Thus, such an intersection type needs to be reviewed by a domain expert to determine whether it is indeed incorrect, and the few concepts have truly been misclassified. This step may be done by a single expert and does not require the involvement of a committee of coordinated domain experts and knowledge engineers.

The methodology presented in this paper has the great advantage that it does not attempt the impossible, namely try to avoid inconsistencies between knowledge engineers. Rather, it capitalizes on the inconsistencies to perform semantic refinement. This stands in considerable contrast to the current methods of knowledge modeling and is a great strength of our approach, because it allows domain experts to work fairly independently on their individual microtheories. The method of semantic refinement presented in this paper highlights sets of complex concepts and makes it easier for domain experts to detect errors during a later review stage.

Clearly, our methodology is very different from Formal Concept Analysis [40,41]. While Formal Concept Analysis treats attributes as first-class citizens that are fully integrated into the representation, our semantic refinement model completely ignores attributes. Supplying a complete, correct, consistent set of attributes for every concept in a large semantic network is a very difficult task which would require a high degree of communication between all the experts involved in the analysis. We are using attributes (and relationships) in our semantic network. However, as these are not integral to the refinement process, the need for a complete specification of attributes is eliminated in our methodology. This reduces the difficulty of building the semantic network in the first place.

Having outlined our methodology of semantic refinement, the question remains: why would anybody go through the major effort of assigning concepts to original semantic types? The reality is that precisely this assignment was made at great expense in the Unified Medical Language System (UMLS) [14,16,20,39] long before we developed the semantic refinement methodology, and without any involvement of our team of researchers. The motivation of the National Library of Medicine (NLM) in developing this classification was to create a high-level abstraction that helps users to orient themselves to the knowledge base, which integrates several existing medical terminologies. Such a high-level abstraction, taking the form of a semantic network, is thus recommended for other knowledge bases as well. We note that our **Step 5** results in a more detailed high-level abstraction for the UMLS.

1.2. The unified medical language system

The UMLS [14,16,20,39], designed by the NLM, combines many well established medical informatics terminologies in a unified knowledge representation system. It consists of three knowledge sources of which we are interested in two, the Metathesaurus and the semantic network [22–24]. The Metathesaurus is a unified collection of many different terminologies (over 100), and it is also the source of the name thesaurus used above in the summary of our methodology.

The UMLS can be used by a wide variety of application programs to overcome the retrieval problems caused by differences in the way the same medical concept is expressed in different sources [15]. Such a resource is valuable to medical researchers and the healthcare industry.

The UMLS is large and complex. The scope and complexity of the UMLS pose serious comprehension problems for users and even developers. The magnitude of presented knowledge is overwhelming for human comprehension capabilities. It is difficult to maintain and use the UMLS without understanding its structure. Designers, maintainers and users of the UMLS need tools to help with their work. There are tools for retrieval and manipulation of the content of the UMLS [6,31,35,37,38]. However, such tools are insufficient. Rather, tools should also help professionals reach a level of *comprehension* essential to performing their tasks.

The semantic refinement methodology presented in this paper was developed to provide this kind of help. Contrary to the development of tools, this methodology works by performing a *structural* improvement of the UMLS. Thus, the methodology results in a better constrained semantics for individual concepts. The work presented in this paper builds on our previous work [21] in which we have attacked complex terminologies and vocabularies both by structural techniques and by tool development. In [9], we described how our previous work on the medical entities dictionary (MED) [7] helped its designer uncover and correct some errors and inconsistencies in the MED's original modeling and improve its contents.

To describe the UMLS in more detail, the Metathesaurus is a compilation of terms, concepts, relationships, and associated information. In the 1998 release of the Metathesaurus, there are 1,051,901 term names mapped into 476,313 concepts.

In the following paragraph we are placing the word ''original'' in parentheses, as a reminder that ''original semantic type'' is our terminology, and not UMLS terminology. In the UMLS these semantic types are just called ''semantic types.''

The semantic network of the UMLS contains (original) semantic types (e.g., **Disease or Syndrome, Virus**). The hierarchy of the semantic network consists of IS-A links. In addition there are non-IS-A relationships among these (original) semantic types (e.g., **Virus** *causes* **Disease or Syndrome**) [22–24]. [2] The 1998 release of the semantic network contains 132 (original) semantic types, 130 IS-A relationships and 53 kinds of non-IS-A relationships. **Entity** and **Event** are the two roots of the hierarchy.

Fig. 1 shows the subnetwork of the semantic network rooted at **Event**. A semantic type is drawn as a rectangle with its name written inside; an IS-A link is a bold arrow directed from a child semantic type to a parent semantic type. A non-hierarchical relationship is represented as a thin labeled arrow. When a relationship is directed to an original semantic type in the **Entity** part of the semantic network (not shown in the figure), then this semantic type is indicated by a circle with a question mark inside. Due to space limitations, some relationship labels in the figure have been replaced by numbers. The legend shows the correspondence between these numbers and the labels.

As will become clear in the context of this paper, we believe that the great investment made in the UMLS by the NLM should be considered a guide to other areas of science and other domains of knowledge. We hope to see the development of similarly ambitious projects in many areas, especially in the different disciplines of engineering. Our methodology will then permit the semantic refinement of those Terminological Knowledge Bases, resulting in better expressed semantics for them.

A preliminary version of our ideas was published in [10]. The presentation in [10] is informal and specific to the UMLS. It relies on very limited mathematical tools. In this paper, we present a complete formal treatment of IS-A link introduction. Furthermore, this paper describes a complete methodology for the development and refinement of perfectly general Terminological Knowledge Bases from scratch. It addresses the problem of terminological knowledge from the perspectives of Artificial Intelligence, expert systems and knowledge engineering. The analysis in [10] was of a specific existing system, namely, the UMLS. The refinement methodology was introduced "after the fact" for checking purposes only, and not as a general methodology of building Terminological Knowledge Bases. Furthermore, the UMLS is limited to use in medical informatics, while the methodology described in this paper is domain independent. Examples in this paper, even though taken from the UMLS, do not require medical knowledge, while the examples in [10] were geared towards readers with a medical background.

In this paper, we are primarily interested in the algorithmic steps (**Step 4** and **Step 5**) of our methodology. Thus, the rest of this paper is organized as follows. Section 2 describes **Step 4**, the algorithmic derivation of pure semantic types and intersection types from a set of original semantic types. As our work was performed in the context of the UMLS, we will present examples based on the UMLS semantic network and Metathesaurus. Section 3 presents **Step 5**, the algorithmic approach of how to specify the IS-A relationships between new semantic types. **Step 6** of our methodology is covered in Section 4 in the context of describing the benefits of the augmented semantic network. Section 5 contains our conclusions.

---

[2] Typographical conventions: Semantic types always appear in bold. The first letters of concepts are always capitalized. Concepts are written between double quotes only if there is a danger of confusion.

Fig. 1. Event subnetwork of the semantic network.

2. Semantic refinement by definition of pure and intersection types

2.1. Classification into original semantic types

Our model of semantic refinement requires that domain experts initially construct a knowledge base that can be algorithmically refined. For this purpose, a model of the domain is constructed which consists of a fairly small number of high-level concepts and of the relationships that connect those concepts. As noted before, the high-level concepts are called original semantic types. The classification of a thesaurus of concepts by the semantic types of a semantic network is defined to be *complete* if every concept of the domain is assigned to at least one semantic type. The classification is defined to be *disjoint* if every concept of the domain is assigned to at most one semantic type.

The gathering of concepts, the introduction of original semantic types, and the assignment of concepts to original semantic types happen in **Steps 1–3** of our methodology. These three steps abstract the process that has gone into the development of the UMLS.

The connection between the semantic network and the (meta)thesaurus has been described in [25] as follows: ''The semantic network encompasses and provides a unifying structure for the thesaurus constituent vocabularies.'' As the semantic network is *much* smaller than the thesaurus, the semantic network provides a high-level, compact abstract view of the thesaurus. The classification of the concepts of the thesaurus by the original semantic types of the semantic network is complete but not disjoint.

2.2. Original semantic types do not classify uniform sets

Now, we need to describe the refinement algorithm of **Step 4**. This algorithm takes the assignment of concepts to *original semantic types* and changes it, creating in the process intersection types while transforming the original semantic types into pure semantic types. As a result of this reassignment of concepts, every concept will be assigned to one and only one semantic type. This classification into new semantic types is both complete and disjoint.

A concept may belong to several original semantic types. Therefore, the set of concepts of one original semantic type may be *non-uniform*. The notion of *non-uniformity* will first be elucidated by an example. We shall show the non-uniformity of the original semantic type **Environmental Effect of Humans**, which has 42 concepts. For an alphabetic list of the concepts of the original semantic type **Environmental Effect of Humans**, see Table 1.

A few of the concepts assigned only to **Environmental Effects of Humans** are Greenhouse Effect, Industrial Smog, Second hand cigarette smoke, and Water Pollution. The concept Fluoridation is classified as both **Environmental Effect of Humans** and as **Therapeutic or Preventive Procedure**. The concept Desertification is classified as **Environmental Effect of Humans** and as **Phenomenon or Process**. The concept Environmental Air Flow is classified as **Environmental Effect of Humans** and as **Human-Caused Phenomenon or Process**. The concepts Acid Rain, Radioactive Fallout, Radioactive Waste, and Smoke are classified as **Environmental Effect of Humans** and as **Hazardous or Poisonous Substance**. The concept Industrial Waste is classified as three original semantic types, **Environmental Effect of Humans**, **Manufactured Object** and **Hazardous or Poisonous Substance**. All these extra assignments of concepts that are also assigned to **Environmental Effect of Humans** are not visible to a user who is looking at Table 1.

Table 1
All concepts assigned to the original semantic type **Environmental Effect of Humans**

Environmental Effect of Humans
Acid Rain
Air Pollution
Air Pollution, Indoor
Air Pollution, Radioactive
Bathing water pollution
Deforestation
Desertification
Drinking water pollution
Dust pollution
Environmental Pollution
Environmental air flow
Exhaust fumes
Fluoridation
Food Contamination, Radioactive
Garbage
Global Warming
Greenhouse Effect
Heating
Inappropriate temperature in local application and packing
Indoor Air Quality
Industrial Waste
Industrial Smog
Noise, Transportation
Oil spill
PBC airborne level
Pollution and pollution exposures
Pollution, NOS
Radioactive Fallout
Radioactive Waste
Second hand cigarette smoke
Sewage
Sludge
Smoke
Smoking, Passive
Soil degradation
Soil pollution
Suburbanization
Tobacco Smoke Pollution
Water Pollution
Water Pollution, Chemical
Water Pollution, Radioactive
Water Pollution, Thermal

It is difficult to comprehend and use the information contained in a non-uniform semantic type such as **Environmental Effect of Humans**. This is the reason why coding systems, such as the Dewey Classification of books or the DRG coding of medical diagnoses, use several levels of classification. These levels may be expressed by a sequence of digits rather than by a single level of very

general classifications, which yields a more refined classification. The multilevel classification helps user orientation to the codes, as codes with long identical prefixes represent closely related information. Furthermore, all items of information falling under any specific code tend to constitute relatively uniform sets.

The problem we face is how to group concepts associated with multiple original semantic types into uniform sets. For this purpose, we need to define the semantics of concepts of the thesaurus in the context of their classification. A natural way to define the semantics of each concept of the thesaurus is to derive it from the semantic network. Hence, the semantics of a concept are partially provided by the original semantic types assigned to it. If a concept is assigned to only one original semantic type, then it has *simple semantics*. Otherwise, if a concept is assigned to a set of original semantic types, it has *compound semantics*, defined by the combination of its different original semantic types. Thus, looking at the above example, the concepts of the semantic type **Environmental Effect of Humans** do not share the same semantics.

For example, the concept Air Pollution has the simple semantics of **Environmental Effect of Humans**, and the concept Fluoridation has the compound semantics of **Environmental Effect of Humans ∩ Therapeutic or Preventive Procedure**. The symbol "∩" indicates the intersection, meaning that the concept Fluoridation is both an environmental effect of humans and a therapeutic or preventive procedure. Fig. 2 is a Venn diagram showing all intersections among six original semantic types, **Environmental Effect of Humans** and the five original semantic types with which it intersects. Each intersection contains concepts that belong to two or more original semantic types. From Fig. 2, we see that all 42 concepts of **Environmental Effect of Humans** are classified into six groups with different semantics. The concepts of one group have simple semantics, while the other five groups express compound semantics.

Now that each concept has its own semantics defined, we can rephrase the vague statement "the set of concepts of one original semantic type may be non-uniform." A set of concepts is *uniform*, if and only if all concepts of the set are assigned to exactly the same set of semantic types. Otherwise it is called non-uniform. The concepts in a non-uniform set have differing semantics. A semantic type is uniform if all the concepts assigned to it form a uniform set. If a semantic type is not uniform, we call it non-uniform.

It is more difficult to comprehend and use a set of concepts of differing semantics. Hence, the challenge is to create an alternative classification system of semantic types, each of which is assigned all and only concepts of the same semantics, either simple or compound. Concepts of simple semantics are assigned to pure semantic types. Concepts of compound semantics are assigned to intersection types.

2.3. Pure semantic types

As a reminder, each pure semantic type is one of the originally given semantic types; however, the set of concepts assigned to it has been stripped down. Only those concepts that were *not* assigned to any other semantic type are still assigned to it. We start by discussing the automatic creation of pure semantic types. Every original semantic type is transformed into a pure semantic type of the same name. Concepts are reassigned in the following way.

We define an operator $\overline{\mathbf{S}}^{O}(C)$ which returns for every concept C the set of all original semantic types that C is assigned to. Every operator $\mathbf{x}$ written as $\overline{\mathbf{x}}$ is considered a set-valued operator.
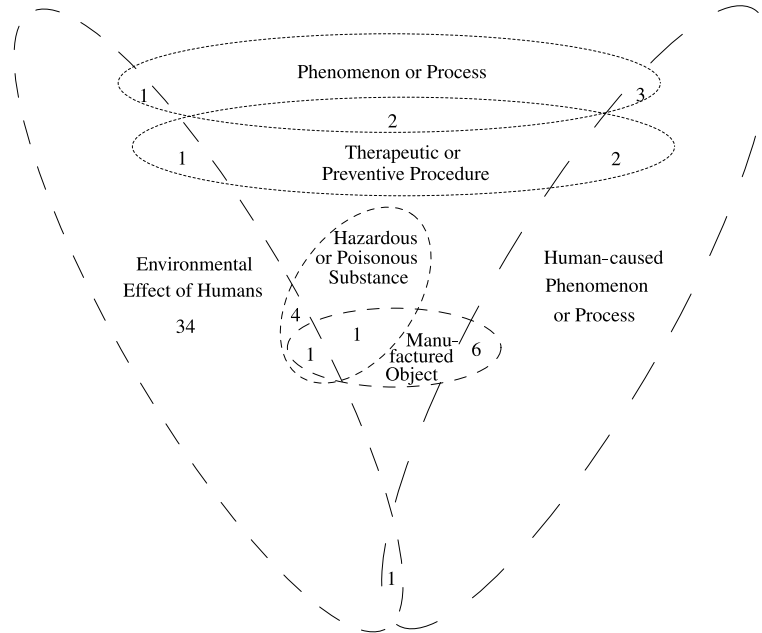
Fig. 2. Venn diagram of the semantic type **Environmental Effect of Humans** and its intersecting semantic types and intersections among them. (Numbers denote cardinalities of intersections).

Thus, there is a scalar version of this operator also: $\mathbf{S}^O(C)$. The scalar version may be used whenever the set-valued operator returns a singleton set. The operator $\overline{\mathbf{S}}^N(C)$ returns the set of new semantic types of C. The corresponding scalar operator $\mathbf{S}^N(C)$ returns the new semantic type of C. Finally, $\mathbf{S}^P(S)$ returns the corresponding pure semantic type of the original semantic type S. This operator is always used in the scalar form only.

It will be noted that $|\overline{\mathbf{S}}^N(C)| = 1$ holds for any C. Also the following mapping condition applies.

$$\forall C[(|\overline{\mathbf{S}}^O(C)| = 1) \Rightarrow \mathbf{S}^N(C) = \mathbf{S}^P(\mathbf{S}^O(C))]$$

In words, if a concept is initially assigned to *only* one original semantic type, will it be assigned to a pure semantic type of the same name after the refinement. For instance, in the UMLS, the concept Air is uniquely assigned to the original semantic type **Substance**. After mapping, Air is assigned to the pure semantic type **Substance**. There are 34 concepts classified by the pure semantic type **Environmental Effect of Humans**. These are listed in Table 2. All 34 concepts have the same semantics of **Environmental Effect of Humans** only. Thus, this set is semantically uniform. A total of 357,804 concepts in the UMLS Metathesaurus are assigned to only one original semantic type.

Rule: Artificial existence for pure semantic types: In the unlikely case that an original semantic type exists that has no concept assigned to it and it alone, there will be no concepts assigned to its corresponding pure semantic type. Nevertheless, the corresponding pure semantic type will be generated to avoid a loss of hierarchical information in the semantic network.

Table 2
Concepts assigned to the pure semantic type **Environmental Effect of Humans** and to its intersection types

Environmental Effect of Humans

Air Pollution
Air Pollution, Indoor
Air Pollution, Radioactive
Bathing Water Pollution
Deforestation
Drinking water Pollution
Dust pollution
Environmental Pollution
Exhaust fumes
Food Contamination, Radioactive
Garbage
Global Warming
Greenhouse Effect
Heating
Inappropriate temperature in local application and packing
Indoor Air Quality
Industrial smog
Noise, Transportation
Oil spill
PBC airborne level
Pollution and pollution exposures
Pollution, NOS
Second hand cigarette smoke
Sewage
Sludge
Smoking, Passive
Soil Degradation
Soil pollution
Suburbanization
Tobacco Smoke Pollution
Water Pollution
Water Pollution, Chemical
Water Pollution, Radioactive
Water Pollution, Thermal

Environmental Effect of Humans ∩ Therapeutic or Preventive Procedure
Fluoridation

Environmental Effect of Humans ∩ Phenomenon or Process
Desertification

Environmental Effect of Humans ∩ Human-caused Phenomenon or Process
Environmental air flow

Environmental Effect of Humans ∩ Hazardous or Poisonous Substance
Acid rain
Radioactive fallout
Radioactive waste
Smoke

Table 2 (*continued*)

Environmental Effect of Humans
Environmental Effect of Humans ∩ **Manufactured objects** ∩ **Hazardous or Poisonous Substance**
Industrial waste

For an example of an application of this rule, see the following subsection.

2.4. Intersection types

If a concept belongs to several original semantic types, it has compound semantics, as defined above. In order to avoid a non-uniform semantics for the set of all the concepts belonging to a semantic type, each concept that is assigned to several original semantic types needs to be reassigned as belonging to a *unique* new semantic type. Thus, an additional kind of semantic type is needed.

In this paper, we call the set of concepts belonging to a semantic type S the extent of S and denote it by $E(S)$. Consider now the reclassification of concepts with compound semantics. Let C be a concept assigned to the k original semantic types $S_{i_1}, S_{i_2}, \ldots, S_{i_k}$ out of all the n semantic types $S_1, S_2, \ldots, S_n$ of the domain. Obviously, the concept C is not assigned to any of the pure semantic types $\mathbf{S}^{\mathrm{P}}(S_{i_1}), \mathbf{S}^{\mathrm{P}}(S_{i_2}), \ldots, \mathbf{S}^{\mathrm{P}}(S_{i_k})$. Since the concept C is assigned to the original semantic types S_{i_j}, $(1 \leq j \leq k)$, we write $C \in E(S_{i_j})$, $(1 \leq j \leq k)$, where $E(S_{i_j})$ is the extent of the original semantic type S_{i_j}. Since this is true for $1 \leq j \leq k$, the concept C belongs to the intersection of the extents $\bigcap_{j=1}^{k} E(S_{i_j})$. Thus, it is natural to define a new kind of semantic type, called an intersection type, which has this intersection as its extent.

For example, the concept C will be assigned to an intersection type S_I which represents the combination of all the original semantic types to which C was assigned, i.e., $S_{i_1}, S_{i_2}, \ldots, S_{i_k}$. Furthermore, all the concepts which belong to the intersection of the extents $\bigcap_{j=1}^{k} E(S_{i_j})$ are assigned to the same intersection type. In this way, all the concepts assigned to the intersection type S_I share the same compound semantics:

$$E(S_I) = \bigcap_{j=1}^{k} E(S_{i_j})$$

In order to create intersection types, all concepts with multiple original semantic types are partitioned into groups such that each group contains all the concepts belonging to the same set of original semantic types. That means the concepts in each group have the same compound semantics (i.e., the set is uniform). After we obtain the groups from the partitioning process, exactly one intersection type is created for each group. Furthermore, the concepts in each group are assigned to the corresponding intersection type.

If $\overline{\mathbf{S}}^{\mathrm{O}}(C_1) = \{S_1, S_2\}$, we use the notation $S_I = S_1 \cap S_2$ for the resulting intersection type. Table 2 shows the extents of several intersection types of the original semantic type **Environmental Effect of Humans** with other semantic types. The symbol "∩" is used in naming the intersection types in the table. It is easier to comprehend the set of concepts of the original semantic type **Environmental Effect of Humans** in Table 2, where the set is divided into extents of semantic types of

Table 3
Concepts assigned to the intersection types between the semantic types intersecting the original semantic type **Environmental Effect of Humans**

Intersections of the semantic types intersecting **Environmental Effect of Humans**
Therapeutic or Preventive Procedure ∩ Phenomenon or Process Feedback Vibration ⟨1⟩
Therapeutic or Preventive Procedure ∩ Human-caused Phenomenon or Process Decontamination Employment, Supported
Phenomenon or Process ∩ Human-caused Phenomenon or Process Nuclear Accidents Nuclear Reactor Accidents Accidents, Radiation
Human-Caused Phenomenon or Process ∩ Manufactured Object Concentration Camps Family Planning, Environment Office Automation Video Recording Videodisc Recording Videotape Recording
Manufactured Object ∩ Hazardous or Poisonous Substance Hazardous waste

uniform semantics, rather than from Table 1, which contains a non-uniform set of concepts. Table 3 shows the extents of intersection types involving those original semantic types that intersect the original semantic type **Environmental Effect of Humans**.

Regarding the naming of intersection types, the list of intersecting semantic types of each intersection type should be reviewed by domain experts to identify simpler names whenever possible. For example, the intersection type **Pharmacologic Substance ∩ Organic Chemical** can be renamed **Organic Pharmacologic Substance**. As another example, the intersection type **Body Part, Organ, or Organ Component ∩ Medical Device** can be renamed **Prosthesis**. If, however, no appropriate name is identified, the intersection symbol "∩" is used to clarify the compound semantics of the type. After the creation of the intersection types, all 476,314 concepts in the UMLS Metathesaurus are assigned to a new semantic type. More precisely, each is assigned to *one* new semantic type in the augmented semantic network. The whole augmented semantic network consists of 1296 types. Of these, 1163 are intersection types.

As an abstract example, assume that the concepts "g" and "h" belong to the original semantic type **W** (Fig. 3). The concepts "a," "b," "d," "e," "f" and "g" belong to the original semantic type **X**, the concepts "b" and "g" belong to the original semantic type **Y**, and the concepts "c," "d," "e" and "g" belong to the original semantic type **Z**. Then our algorithm will transform **W** into a pure semantic type with one assigned concept "h." **X** is transformed into a pure semantic type with two assigned concepts "a" and "f." **Y** has two concepts, "b" which is shared with **X**, and "g," which is shared with **Z**. Thus, there are no concepts left for the pure

Before applying the Algorithm
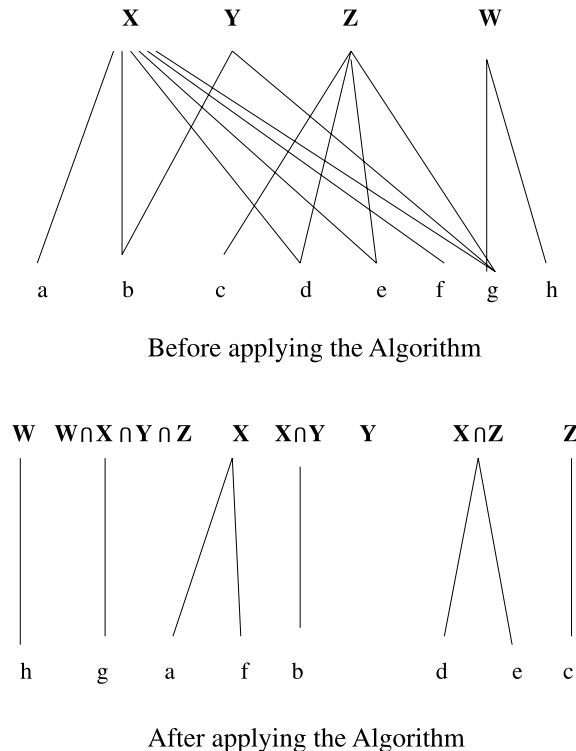

After applying the Algorithm

Fig. 3. An example for the semantic refinement process.

semantic type **Y**. However, by the artificial existence rule, **Y** is nevertheless kept as a pure semantic type with no concepts. **Z** will be transformed into a pure semantic type with one assigned concept "c."

Next, our algorithm will construct three intersection types. The concept "g" belongs to **W**, **X**, **Y** and **Z**. As such, it induces an intersection type $W \cap X \cap Y \cap Z$. The concept "b" belongs to **X** and **Y** and induces an intersection type $X \cap Y$ that has only the concept "b" assigned to it. Similarly, "d" and "e" induce an intersection type $X \cap Z$. We note that there is no concept that belongs to only **Y** and **Z**, thus there is no intersection type $Y \cap Z$.

All the mappings described in this subsection and the previous subsection are easily implemented, and indeed, they have been applied to the UMLS. Thus, we will now show corresponding examples from the UMLS.

It may seem that with the intersection types, we lose access to the extents of the original semantic types. However, in the next section, we will show that this information can easily be reconstructed upon demand.

3. The IS-A relationships of the augmented semantic network

Having identified all the new semantic types, we need to connect them with appropriate IS-A links. In [10], we discussed a method of IS-A link introduction which results in what we call the

one-level extension network. In this method, an intersection type is connected to all the pure semantic types that it is an intersection of. Thus, the intersection type will be one level lower in the semantic network than its lowest pure semantic type parent. As a result, the original semantic network is extended by one level, which explains the name of this network. For examples, see [10].

In the following discussion, we use the expressions "*A* IS-A *B*" and "*A* is a child of *B*" interchangeably. The one-level extension network typically has more IS-A relationships than are absolutely necessary. By using the transitivity of the IS-A relationship, it becomes possible to reduce the average number of parents for intersection types with more than two parents. (Every intersection type has at least two parents.) If an intersection type *A* with $n > 2$ parents exists, and an intersection type *B* has the same *n* parents plus one additional parent, then *B* may point with one IS-A link to *A* instead of pointing to all the *n* parents of *A*.

In the following example, we will use these abbreviations:

E = **Environmental Effect of Humans**,

H = **Hazardous or Poisonous Substance**,

and

M = **Manufactured Object**.

For example, in Fig. 2, we see the intersection type $\mathbf{E} \cap \mathbf{M} \cap \mathbf{H}$ which, by the above method, is a child of three pure semantic types **E**, **M** and **H**. Similarly, the intersection type $\mathbf{E} \cap \mathbf{H}$ is a child of **E** and **H**. If we compare these two intersection types, we see that the semantics of $\mathbf{E} \cap \mathbf{M} \cap \mathbf{H}$ is more specific than the semantics of $\mathbf{E} \cap \mathbf{H}$. Similarly, the semantics of $\mathbf{E} \cap \mathbf{H}$ is more specific than the semantics of **E**. Hence, it is natural to have an IS-A relationship from the more specific intersection type $\mathbf{E} \cap \mathbf{M} \cap \mathbf{H}$ to the more general intersection type $\mathbf{E} \cap \mathbf{H}$ rather than to **E** and to **H**.

We will now explain why the resulting IS-A configuration is mathematically correct. Since we are modeling $\mathbf{E} \cap \mathbf{M} \cap \mathbf{H}$ as child of $\mathbf{E} \cap \mathbf{H}$, and $\mathbf{E} \cap \mathbf{H}$ is a child of **E** and of **H**, the transitivity of the IS-A relationship implies that $\mathbf{E} \cap \mathbf{M} \cap \mathbf{H}$ is a child of **E** and of **H**. Thus, there is no need to have an explicit IS-A relationship from $\mathbf{E} \cap \mathbf{M} \cap \mathbf{H}$ to **E** or to **H**.

In view of this example, we will discuss a method of IS-A link introduction which results in a *multi-level extension network* (Fig. 4). The multi-level extension network is designed to capture semantic relationships between intersection types which were not reflected by the one-level extension method. We allow an intersection type to be a child of another intersection type. As a result, intersection types appear in multiple levels.

In order to systematically define the IS-A relationships of intersection types, we need a rule to determine the parents of each intersection type. Before we describe such a rule, we first need to give the definitions of (1) the maximal subsets of a set and (2) the minimal supertypes of an intersection type.

Let *U* be a universal set of elements and let *F* be a given family of sets over *U*. That is, *F* is a set of sets. Every set in *F* contains elements from *U*. In other words, *F* is a subset of the power set of *U* ($F \subset 2^U$).

As before, $E(S)$ stands for the extent of a semantic type *S*. In the context of the UMLS, the universal set *U* is the set of all concepts of the Metathesaurus, and the family *F* is the family of the extents of all semantic types in the semantic network.
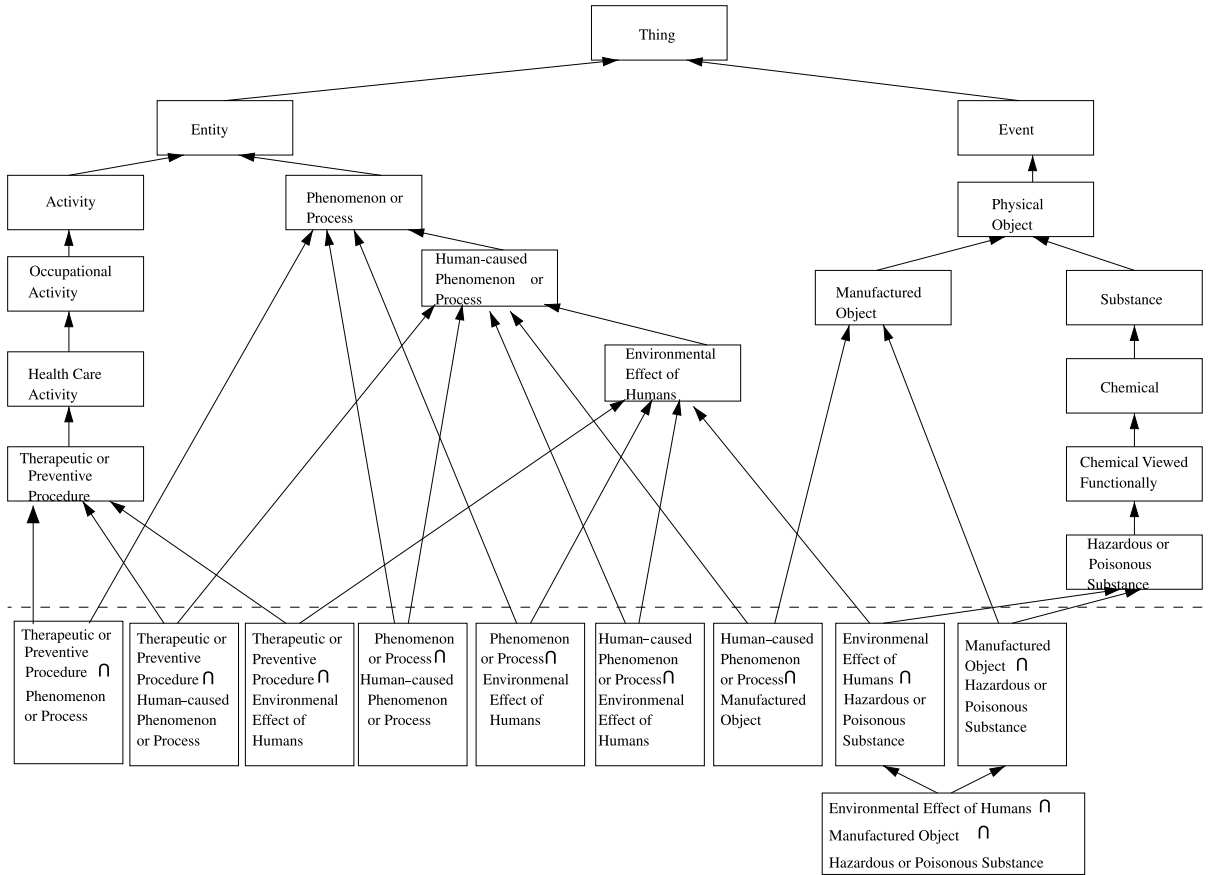
Fig. 4. A subnetwork of the UMLS augmented semantic network using the multi-level extension method.

$$F = \{E(S_1), E(S_2), \ldots, E(S_n)\};$$

n is the number of semantic types. In this section, whenever we refer to extents, these are extents of semantic types. For a given family of extents G $(G = \{E(S_{i_1}), E(S_{i_2}), \ldots, E(S_{i_k})\}, (k < n))$ which is a subfamily of F, the *family intersection* I_G is the intersection of all extents in G. ($I_G = \bigcap_{1 \leq j \leq k} E(S_{i_j})$.) The intersection type corresponding to the family intersection I_G is denoted by S_{I_G}.

When an intersection type is given, it is possible to identify all its *potential parents* for which there may exist an implied IS-A relationship. E.g., for the intersection type S_{I_G}, each of the semantic types S_{i_j} $(1 \leq j \leq k < n)$ is a potential parent of S_{I_G}. Furthermore, for each family D of extents with a non-empty family intersection I_D, such that D is a subfamily of G, the intersection type S_{I_D} is a potential parent of S_{I_G}.

Definition 1 (*Maximal Subset*): Let A and B be sets that are elements of F, such that A is a proper subset of B. If there does not exist a set C in F such that A is a proper subset of C and C is a proper subset of B, then we call A a *maximal subset* of B in F.

E.g., if $\{X, Y, Z\}$, $\{X, Y\}$, and $\{X\}$ are three sets in a family F, $\{X, Y\}$ is a maximal subset of $\{X, Y, Z\}$. But $\{X\}$ is not a maximal subset of $\{X, Y, Z\}$, since $\{X\} \subset \{X, Y\}$ and $\{X, Y\} \subset \{X, Y, Z\}$. So, $\{X, Y\}$ plays the role of an intermediate subset between $\{X\}$ and $\{X, Y, Z\}$.

Since a family is a set of sets, the definition of "maximal subset" can be modified appropriately to define a maximal subfamily. Now we define the notion of minimal supertype of an intersection type, corresponding to the above definition of a maximal subfamily of a family.

Definition 2 (*Minimal Supertype*): Let S_{I_G} be an intersection type corresponding to the family intersection I_G. If S_{I_D} is a potential parent of S_{I_G}, then S_{I_D} is a minimal supertype of S_{I_G} if and only if the family D is a maximal subfamily of the family G.

Intuitively, a minimal supertype of an intersection type S is a parent that is most similar to S. As such, it has the potential to maximize the number of semantic types to which S does not need a direct IS-A link.

For example, the intersection types $\mathbf{E} \cap \mathbf{H}$ and $\mathbf{M} \cap \mathbf{H}$ are the only two existing minimal supertypes of the intersection type $\mathbf{E} \cap \mathbf{M} \cap \mathbf{H}$. There are no concepts that belong to the intersection type $\mathbf{E} \cap \mathbf{M}$, and thus such an intersection type does not exist.

Note that as a special case, D may be a family of the extent of only one semantic type, such as **Environmental Effect of Humans**. Denote this semantic type by S_i. In this case, S_{I_D} degenerates to an original semantic type S_i rather than an intersection type. We have chosen to simplify the above definition by not explicitly considering this special case. Assuming that S_i is a degenerate intersection type, our definition is still applicable.

Rule: Multi-level extension: Let S_I be an intersection type in the network. Then IS-A relationships are defined from S_I to all its minimal supertypes in the network.

This rule is guaranteed to increase the depth of the network by transforming intersection types of more than two semantic types into children of other intersection types. As McCray [25] notes, it is considered desirable to increase the depth of the semantic network.

The multi-level extension network is semantically more accurate than the one-level extension network, as it captures IS-A relationships between intersection types. Furthermore, it typically reduces the number of IS-A links compared to the one-level extension network. As such, it reduces the complexity of the network, while maintaining all necessary relationships among the new semantic types. (We have formally defined the complexity of a network as the ratio of the number of relationships to the number of concepts [11,12,29].)

Unfortunately, there is no guarantee that the multi-level extension network will always contain fewer IS-A links than the one-level extension network. Nevertheless, in our experience, the multi-level extension network typically has fewer IS-A links and deeper networks than the one-level extension network. Both these phenomena are considered desirable. (For an example, see [10].)

Table 4 shows some details of the level distribution of the new semantic types of the multi-level extension network of the UMLS. We obtain a network with depth 14. To summarize, to create the multi-level extension network, we created 1163 intersection types and added 2677 new IS-A relationships. Each of the 476,314 concepts in the Metathesaurus is assigned to a unique new semantic type. The network contains 1296 new semantic types.

Table 4
Number of types in each level of the multi-level extension network of the UMLS

Level	Number of pure semantic types	Number of intersection types
1	1	0
2	2	0
3	4	0
4	20	0
5	41	56
6	23	203
7	23	163
8	17	186
9	2	234
10	0	212
11	0	89
12	0	16
13	0	3
14	0	1

Fig. 4 is a subnetwork of the resulting multi-level extension for the UMLS, which shows the semantic type **Environmental Effect of Humans**, its intersecting types, all their ancestors and all their intersection types. It contains 16 pure semantic types and 10 intersection types distributed over nine levels. A unique, artificial root **Thing** has been inserted in the figure as the parent of **Entity** and **Event**.

In Section 2.4, we mentioned an apparent loss of information caused by our improved modeling, regarding the extents of the original semantic types. To recover the extent of an original semantic type, we combine the extent of the corresponding pure semantic type with the extents of all its intersection type descendants (which are defined with regards to the IS-A relationships).

4. Advantages of the augmented semantic network

4.1. Classification into uniform new semantic types

As we discussed before, the set of concepts belonging to an original semantic type of the UMLS is not necessarily semantically uniform since some of the concepts may belong to additional original semantic types. It is difficult for a user to comprehend and use the set of concepts of such an original semantic type due to this lack of uniformity. Because all concepts in the thesaurus are either assigned to a unique intersection type or to a unique pure semantic type, all extents of the new semantic types are now semantically uniform and therefore easier to comprehend.

The contribution of semantic uniformity of the extents of semantic types to their comprehensibility will be demonstrated repeatedly in Section 4.3. There, we will expose various errors which became easy to detect in the semantically uniform extents of Tables 2 and 3. Those errors were much harder (or impossible) to detect in the original Table 1, because Table 1 shows the semantically non-uniform extent of an original semantic type.

4.2. Reduced average extent size

In the UMLS, the sets of concepts of many original semantic types are very large and hard to comprehend. In the 1998 version, on average, every original semantic type is assigned to about 5000 concepts, and some of them are assigned to many more.

Our semantic refinement algorithm reduces the average number of concepts per pure semantic type to about 2700. The average number of concepts of an intersection type is about 100, which is comparatively small. Having a semantic network with a reduced average number of assigned concepts per new semantic type simplifies the comprehension of each such set.

4.3. Exposing problems in Terminological Knowledge Bases

Representing the intersection types and their assigned concepts enables researchers to study the compound semantics of the concepts of such intersection types. Due to their compound semantics, those concepts are more complex in their nature than concepts of simple semantics. Complex concepts are also more error-prone due to their multiple classifications, most likely done by different domain experts.

Thus, in reviewing the extents of intersection types, it is easier to identify modeling problems than with original semantic types. Similarly, it is easier to identify modeling problems in reviewing the extents of pure semantic types rather than the extents of original semantic types. This is due to the uniform simple semantics of pure semantic types, which helps to highlight concepts that do not fit this semantics. However, the frequency of problems in pure semantic types appears to be lower than in intersection types.

One might think that the increase in the number of semantic types would make error checking more difficult. However, the opposite is the case, as will be explained now. The issue is *not* how many semantic types one has to review, but how many *concepts* need to be reviewed by a human expert. Our methodology makes it easy to identify the most "suspicious" concepts, and it also limits the number of concepts that are designated as suspicious. We consider it unrealistic to hope that all concepts in a network of several hundred–thousand concepts can be reviewed with equal effort (or at all). Our methodology indicates how to prioritize the efforts expended on checking errors and problems.

To test our assumption that moderately sized sets of concepts, assigned to a single intersection type, are less likely to indicate a problem than small sets, we performed the following experiment. A domain expert analyzed the concepts of five randomly chosen intersection types, e.g. **Finding ∩ Health Care Activity**, each with twenty concepts. Out of these 100 concepts, only three were judged by the domain expert to be classification errors.

We performed a statistical analysis of the frequencies of the concept sets assigned to intersection types of different sizes (Table 5). For example, there are 421 intersection types with only one associated concept. If we consider all the intersection types with between one and five associated concepts suspicious, then we need to check only 1456 concepts. This is a very small number of concepts compared to the size of the whole thesaurus.

All the concepts of an intersection type have the same semantics. This helps to detect concepts which do not fit the uniform semantics and therefore may be errors. Thus, it is desired that a human expert should check all concepts of one intersection type in one uninterrupted working

Table 5
Partial distribution of concepts over intersection types

Number of concepts in the intersection types	Number of intersection types
1	421
2	147
3	102
4	65
5	35
6	41
7	32
8	15
9	13
…	…
…	…
3947	1
4582	1
6705	1
19349	1
41564	1

session, capitalizing on the semantic uniformity, in order to build up his own mental context. We note that this is typically possible. The average number of concepts per intersection type is 101.8. Only 70 intersection types have more than 114 associated concepts. Only 41 intersection types have 300 or more concepts. Thus, the process of checking concepts is also better modularized than is the case for the original semantic types.

Finally, there are intersection types that have so many assigned concepts that it is impossible to analyze all of them with a reasonable effort. We can still easily verify whether such intersection types make sense, since we know exactly for every intersection type what it is an intersection of. For example, we found an intersection type (with a small extent) **Human-Caused Phenomenon or Process** ∩ **Manufactured Objects** which we judged to be highly suspicious. However, no suspicious intersection type with a large extent was found. All the large intersection types represent reasonable combinations. For example, the intersection of **Organic Chemical** and **Pharmacologic Substance** is the intersection type with the largest number of concepts, namely, 41,564.

In the following subsections, we will discuss several modeling problems. We stress that this is not "pure theory." We have identified actual modeling problems in our previous work [9] with the MED [7] and our previous work [10] with the UMLS. Examples in this section will be drawn from the UMLS.

4.3.1. Ambiguity

An intersection type may highlight a case of an overlooked homonym. Up to this point, we have assumed that there are no homonyms in the initial specification of concepts. If a term indeed is commonly used for two or more concepts, then domain experts have to create distinguishable lexical items for each concept. For example, the UMLS distinguishes between two different senses of a term T by different lexical items like $T\langle 1 \rangle$ and $T\langle 2 \rangle$.

However, the assumption that different senses of each term are recognized from the beginning, and that different experts agree on which sense is represented by which lexical item, is overly optimistic, given the great degree of autonomy which we allow each expert. Thus, terms used in the assignment of concepts to original semantic types might really be ambiguous. Such ambiguous concepts would tend to belong to pairs of very different original semantic types. As a result of this, "strange" intersection types would be generated.

A domain expert could recognize such a problem and disambiguate such concepts by using different terms for the different senses. If all the concepts of an intersection type are disambiguated in this way, then there is no more need for such an intersection type.

As an example, the intersection type **Human-Caused Phenomenon or Process ∩ Manufactured Objects** contains the concepts Concentration Camps, "Family Planning, Environment," Office Automation, Video Recording, Videodisc Recording and Videotape Recording. This intersection type exhibits an obvious contradiction. *The distinction between a (manufactured) object and a (human-caused) phenomenon or process is at a very high level. There are, in all likelihood, no concepts that belong to two semantic types that are so different.* Thus, "Video Recording" appears to be an ambiguous term that really stands for two different concepts: Video Recording ⟨1⟩ is the process of recording something on a video tape. Video Recording ⟨2⟩ is the recorded tape which is an object manufactured as the result of the process of recording. Similarly, a Concentration Camp ⟨1⟩ is a phenomenon relevant because of its medical and psychological impact on survivors of a concentration camp. On the other hand, Concentration Camp ⟨2⟩ refers to a physical environment which is an object created by humans. All six concepts belonging to this intersection type exhibit the same kind of ambiguity, thus all of them are homonyms. After correctly reassigning these (pairs of) homonyms, the intersection type **Human-caused Phenomenon or Process ∩ Manufactured Objects** will disappear. In [10], other examples of such ambiguities in the UMLS are given.

4.3.2. Non-uniform classification

If a domain expert studies the intersection types in comparison with the corresponding pure semantic types, he might find that some of them indicate a non-uniform classification. In non-uniform classifications, some concepts are not assigned to the same original semantic types as other concepts to which they are similar in nature. This is an unacceptable situation, which can be fixed in two ways. Either we assign all the concepts of a similar nature to a relevant original semantic type, or we do not assign any one of the concepts to it. Both these approaches would lead to a uniform classification, as desired.

For example, in the UMLS the concepts Acid Rain, Radioactive Fallout, Radioactive Waste and Smoke are assigned to the intersection type **Environmental Effect of Humans ∩ Hazardous or Poisonous Substance**. Strangely enough, "Water Pollution, Radioactive," "Food Contamination, Radioactive" and "Air Pollution, Radioactive" are assigned to **Environmental Effect of Humans** only. This is clearly a case where uniform criteria need to be applied during modeling. These three concepts need to be reassigned to the intersection type **Environmental Effect of Humans ∩ Hazardous or Poisonous Substance**.

The concept Desertification is the only concept in the intersection type **Environmental Effect of Humans ∩ Phenomenon or Process**. This also appears to be a case of non-uniform classification. Indeed, a review of the concepts classified only under **Environmental Effect of Humans** reveals the

two concepts Deforestation and Suburbanization which rightfully should belong to this intersection type. Interestingly enough, we will see below (Section 4.3.5) that this intersection type itself is redundant. Thus, all three concepts will be moved again.

Domain experts will typically encounter such cases while reviewing intersection type extents. They will identify concepts which are in need of additional assignments to original semantic types. Then a re-execution of the semantic refinement algorithm would correct the extents of such intersection types.

The occurrence of non-uniformities is not surprising when considering that our semantic refinement methodology allows individual experts considerable autonomy. Especially for the UMLS, many experts were involved in the assignment of concepts to semantic types. However, our semantic refinement methodology makes it easier to discover such non-uniformities. Precise information about these non-uniformities should be communicated back to the domain experts who should try to change the original classifications to be more uniform.

4.3.3. Classification errors

Intersection types highlight some classification errors. Such errors may be exposed when the extents of intersection types are reviewed by a domain expert. We did not find a classification error in the example domain of this paper. However, in previous work on the UMLS [10], we noticed that the concept Scotch Tape Mount was assigned to the intersection type **Bacterium ∩ Laboratory Procedure**. This is a strange intersection type, and Scotch Tape Mount should be assigned to **Laboratory Procedure** only. Because Scotch Tape Mount was the only concept assigned to this intersection type, the intersection type itself may be eliminated. In [10], several other examples of classification errors in the UMLS are presented.

As we see from the above example and the others in [10], it is especially beneficial to have experts review very small intersection type extensions. The review may expose that the few concepts were associated with the intersection type by mistake, due to an erroneous classification as an original semantic type. In such a case, the limited effort of a reviewer, who has to inspect only a few concepts, may result in the correction of the originally incorrect classification and the elimination of an intersection type from the augmented semantic network.

4.3.4. Omissions

The extents of intersection types will give the knowledge engineers in charge of the maintenance of a large Terminological Knowledge Base a useful view to discover omissions on the concept level. For example, in the UMLS augmented semantic network, there is an intersection type **Human-caused Phenomenon or Process ∩ Manufactured Object**. This intersection type has six concepts assigned to it: Concentration Camps, ''Family Planning, Environment,'' Office Automation, Video Recording, Videodisc Recording and Videotape Recording. We recognize that a concept Audiotape Recording is omitted, while the concept Audiotape itself does exist in the Metathesaurus. Hence, such a concept should be added to the Metathesaurus and to the intersection type.

A single concept may involve more than one problem. Once we have identified the missing concept Audiotape Recording, it still needs to be disambiguated. As noted before with reference to Videotape Recording, Audiotape Recording may refer to the process of recording as well as to the result of the process.

4.3.5. Redundant classifications

In all the problems discussed so far, a domain expert was needed to recognize a problem and to make the final decision about which kind of problem it was and how to solve it. In this subsection, we will discuss a problem which can be solved algorithmically.

Domain experts are assigning concepts to original semantic types without consulting each other. In some cases, the same concept may be assigned to several original semantic types that are standing in an ancestor–descendant relationship or in a parent–child relationship in the semantic network. However, due to the transitivity of the IS-A relation, it is never necessary to assign a concept to a semantic type A and a parent or ancestor of A. The second assignment is implicit in the first assignment. Furthermore, as the parent or ancestor of A is less specific than A, the semantic constraint achieved by the assignment to the ancestor is much weaker than the semantic constraint achieved by the assignment to A. Thus, whenever such a redundant assignment exists in a network, it needs to be removed. We call such a redundant assignment a *redundant classification*. Redundant classifications can be uncovered by algorithmic means.

For example, in Fig. 4 the intersection type **Phenomenon or Process ∩ Human-caused Phenomenon or Process** has two parents **Phenomenon or Process** and **Human-caused Phenomenon or Process**. However **Human-caused Phenomenon or Process** is itself a child of **Phenomenon or Process**. Therefore, the intersection type **Phenomenon or Process ∩ Human-caused Phenomenon or Process** may be removed from the semantic network without any loss of information. We have identified several other examples of redundant classifications, e.g., **Phenomenon or Process ∩ Environmental Effect of Humans** and **Environmental Effect of Humans ∩ Human-caused Phenomenon or Process**. Fig. 5 shows the revised subnetwork.

We note that redundant classifications should not occur in the UMLS, even according to its designers, as stated in [25]: "In all cases the most specific semantic type available in the hierarchy is assigned to a term." We analyzed the UMLS for redundant classifications and detected 8622 concepts that were assigned to several semantic types standing in ancestor–descendant (or parent–child) relationships. These concepts were reported to the NLM. When all those redundant classifications are removed from the UMLS, 77 intersection types disappeared from the augmented semantic network. Thus, the structure of the augmented semantic network became simpler, without any loss of information.

4.3.6. Independently listed synonyms

While looking at Table 2, we could not avoid noticing that the three concepts (1) "Second hand cigarette smoke," (2) "Smoking, Passive" and (3) "Tobacco Smoke Pollution" appear to be the same concept. Thus, one of the terms should be chosen as the primary concept, and the other terms should be made its synonyms. For example, the two terms "Second hand cigarette smoke" and "Smoking, Passive" are really synonyms for the concept "Tobacco Smoke Pollution."

4.4. Problems in a sample of intersection types

In [10], an arbitrary sample of 100 intersection types from the UMLS with only one assigned concept per intersection type was analyzed by a domain expert. He found that for 11 intersection types, the classification of concepts was correct. For 55 intersection types, the multiple classifications
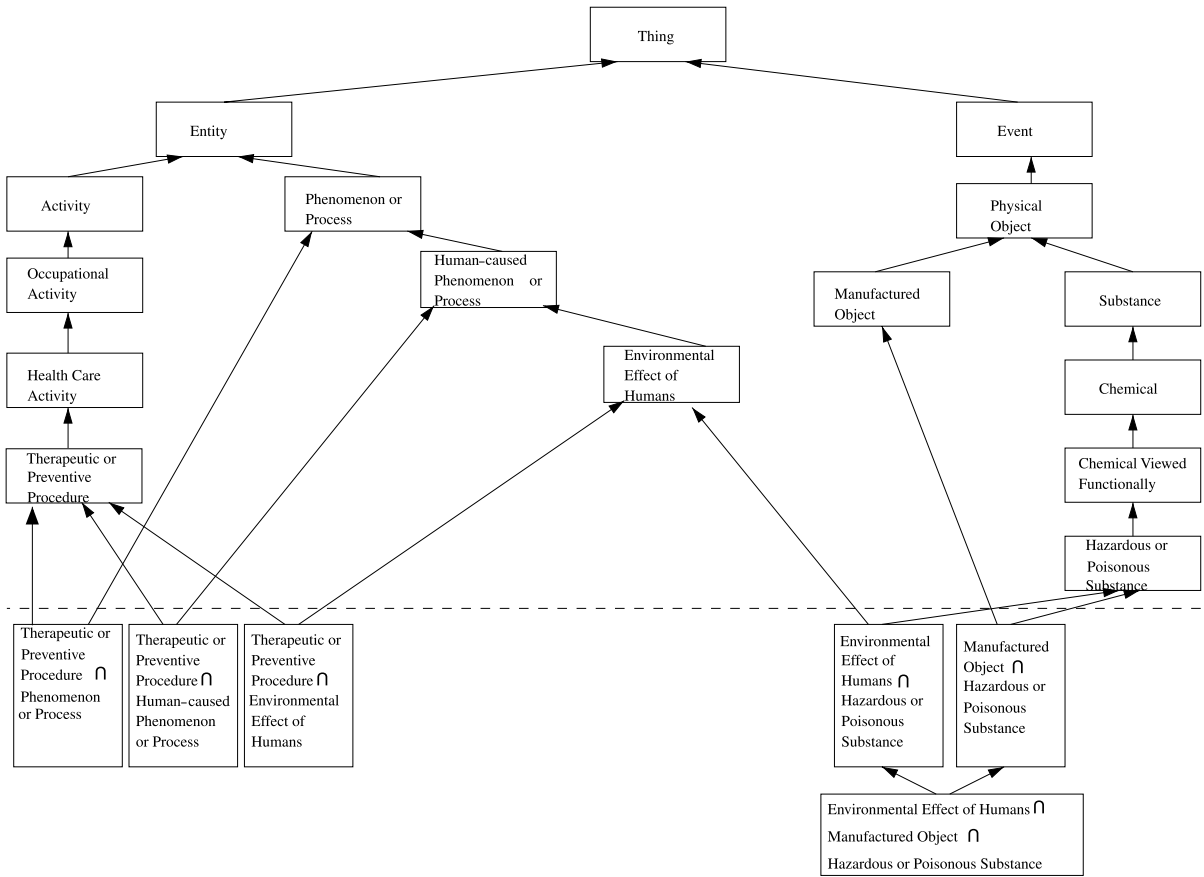
Fig. 5. The subnetwork corresponding to Fig. 4 after removing redundant classifications.

were wrong. For 32 intersection types the classified concepts indicated non-uniform classifications. Finally, two intersection types were redundant classification cases. The results of the analysis of this sample show the potential for finding and correcting many errors and inconsistencies in a network with a limited effort expended on screening intersection types with small extents.

4.5. Extent of the resulting semantic type "Environmental Effect of Humans"

Introducing intersection types has helped us identify a number of errors concerning concepts assigned to the semantic type **Environmental Effect of Humans**, which were discussed in detail in the previous subsections. As a result, we now present the revised version of Table 2 as Table 6 and the revised version of Table 3 as Table 7.

Seven concepts do not appear in Table 7 as they are not assigned to intersection types anymore. Fig. 6 shows the correspondingly revised Venn diagram of Fig. 2. The number of intersection types is down to six, from ten in Fig. 2. The changes in the tables and diagram demonstrate a

Table 6
Concepts assigned to the pure semantic type **Environmental Effect of Humans** and to its intersection types after correction

Environmental Effect of Humans
Air pollution
Air pollution, Indoor
Bathing Water Pollution
Deforestation
Desertification
Drinking Water Pollution
Dust Pollution
Environmental Air Flow
Environmental Pollution
Exhaust fumes
Garbage
Global Warming
Greenhouse Effect
Heating
Inappropriate temperature in local application and packing
Indoor Air Quality
Industrial Smog
Noise, Transportation
Oil spill
PBC airborne level
Pollution and pollution exposures
Pollution, NOS
Second hand cigarette smoke, (Synonyms: ''Tobacco Smoke Pollution,'' ''Smoking, Passive'')
Sewage
Sludge
Soil Degradation
Soil pollution
Suburbanization
Water Pollution
Water Pollution, chemical
Water Pollution, thermal
Environmental Effect of Humans ∩ Therapeutic or Preventive Procedure
Fluoridation
Environmental Effect of Humans ∩ Hazardous or Poisonous Substance
Acid Rain
Air Pollution, Radioactive
Food Contamination, Radioactive
Radioactive Fallout
Radioactive Waste
Smoke
Water Pollution, Radioactive
Environmental Effect of Humans ∩ Manufactured objects ∩ **Hazardous or Poisonous Substance**
Industrial waste

Table 7
Concepts assigned to the intersection types of the semantic types intersecting with the original semantic type **Environmental Effect of Humans**, after correction

Intersections of the semantic types intersecting **Environmental Effect of Humans**
Therapeutic or Preventive Procedure ∩ Phenomenon or Process
Feedback
Vibration ⟨1⟩
Therapeutic or Preventive Procedure ∩ Human-caused Phenomenon or Process
Decontamination
Employment, supported
Manufactured Object ∩ Hazardous or Poisonous Substance
Hazardous waste
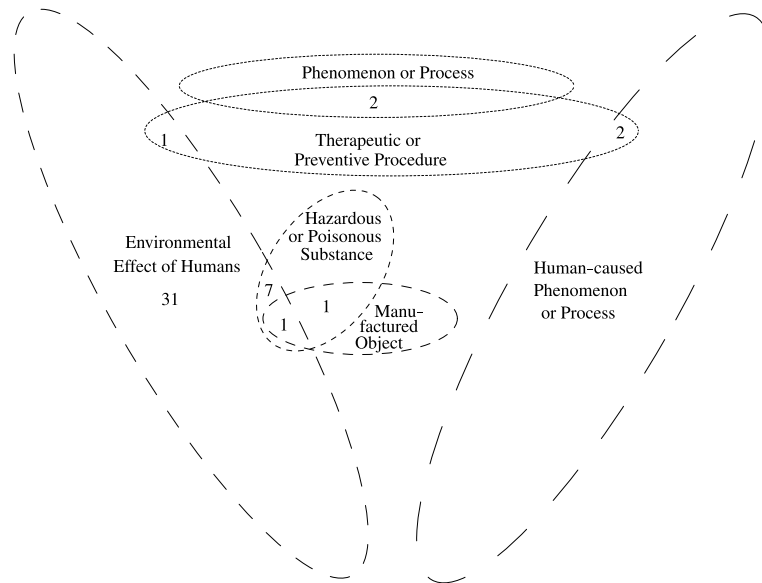
Fig. 6. Revised Venn diagram of the semantic type **Environmental Effect of Humans** and its intersecting types and intersections among them, after correction.

reduction in complexity due to the corrections of errors identified in the extent of the semantic type **Environmental Effect of Humans**.

5. Conclusions

We have developed a methodology that allows us to refine the semantics of concepts in a thesaurus, although it does not result in a unique semantics for each concept. The initial as-

sumption of this methodology is that the semantics of a concept is expressed by assigning the concept to one or more original semantic types. Then we apply an algorithm which, corresponding to unique combinations of original semantic types, identifies intersection types, and assigns concepts to them. All concepts that belong to only one of the original semantic types are assigned to pure semantic types. We also define the augmented semantic network, using the multi-level extension rule.

Our methodology results in unique assignments of concepts to new semantic types which have smaller and semantically uniform extents compared to the original semantic types. Thus, the augmented semantic network expresses a semantic refinement compared to the original semantic network. The result of our methodology is then fed back to domain experts, who can recognize various errors.

Thus, our methodology leads to better Terminological Knowledge Bases. End-users of terminological knowledge have no direct contact with the methodology, but they benefit from its fruits by encountering fewer errors when using a Terminological Knowledge Base. Experience with four domain experts indicates that domain experts consider the task of selecting parts of the UMLS for inspection, and then checking them, manageable. Using our methodology, a large number of modeling errors were identified in the UMLS. We have submitted these errors to the NLM for review. The process of selecting and checking is much more difficult for the UMLS without our methodology. A formal study to quantify these differences is part of future work.

Our methodology has a number of advantages:

- Domain experts involved in building a Terminological Knowledge Base have a single, fairly well defined task, namely, to assign concepts to high level semantic types. They do not have to know about attributes, relationships, *etc.*
- The amount of time during which domain experts have to interact with knowledge engineers is greatly reduced.
- Communication between domain experts is limited to a single step of the six-step methodology, namely, the design of the semantic network and its original semantic types.
- The heart of the refinement methodology is algorithmic.
- The methodology makes it easier to identify errors in a large Terminological Knowledge Base.

The semantic refinement methodology encourages us to believe that the great investment made in the UMLS by the NLM was worthwhile. Similar efforts should be made in other areas of science and other domains of knowledge to build Terminological Knowledge Bases. Our methodology will then allow the same steps of semantic refinement to be applied in these other areas.

Acknowledgements

References

[1] K. Barker, B. Porter, P. Clark, A library of generic concepts for composing knowledge bases, in: First International Conference on Knowledge Capture, K-CAP 01, Victoria, British Columbia, Canada, October, 2001.

[2] L.W. Barsalou, Intraconcept similarity and its implications for interconcept similarity, in: S. Vosniadou, A. Ortony (Eds.), Similarity and Analogical Reasoning, Cambridge University Press, New York, NY, 1989, pp. 76–121.

[3] F.W. Bergmann, J.J. Quantz, Parallel propagation in the description-logic system flex, in: J. Geller, H. Kitano, C.B. Suttner (Eds.), Parallel Processing for Artificial Intelligence, vol. 3, North-Holland, New York, 1997, pp. 181–207.

[4] A. Borgida, R.J. Brachman, D.L. McGuinness, L.A. Resnick, CLASSIC: A structural data model for objects, in: Proceedings of the 1989 ACM SIGMOD International Conference on the Management of Data, appeared as SIGMOD, vol. 18, 1989, pp. 58–67.

[5] R.J. Brachman, H.J. Levesque. The tractability of subsumption in frame based description languages, in: Proceedings of AAAI-84, pages 34–37, Austin, TX, 1984.

[6] K.E. Campbell, D.E. Oliver, E.H. Shortliffe, The unified medical language system: toward a collaborative approach for solving terminologic problems, JAMIA 5 (1) (1998) 12–16.

[7] J.J. Cimino, P.D. Clayton, G. Hripcsak, S. Johnson, Knowledge-based approaches to the maintenance of a large controlled medical terminology, JAMIA 1 (1) (1994) 35–50.

[8] D. Gries, The Science of Programming, Springer-Verlag, New York, 1981.

[9] H. Gu, M. Halper, J. Geller, Y. Perl, Benefits of an OODB representation for controlled medical terminologies, JAMIA 6 (4) (1999) 283–303.

[10] H. Gu, Y. Perl, J. Geller, M. Halper, L. Liu, J.J. Cimino, Representing the UMLS as an OODB: Modeling issues and advantages, Journal of the American Medical Informatics Association 7 (1) (2001) 66–80.

[11] H. Gu, Y. Perl, J. Geller, M. Halper, M. Singh, A methodology for partitioning a vocabulary hierarchy into trees, Artificial Intelligence in Medicine 15 (1) (1999) 77–98.

[12] H. Gu, Y. Perl, M. Halper, J. Geller, F. Kuo, J.J. Cimino, Partitioning an object-oriented terminology schema, Methods in Medical Informatics 40 (2001) 204–212.

[13] S. Harnad, Minds, machines and Searle, Journal of Theoretical and Experimental Artificial Intelligence 1 (1989) 5–25.

[14] B.L. Humphreys, D.A.B. Lindberg, Building the unified medical language system, in: Proceedings of the Thirteenth Annual Symposium on Computer Applications in Medical Care, Washington, DC, November 1989, pp. 475–480.

[15] B.L. Humphreys, D.A.B. Lindberg, The unified medical language system project: a distributed experiment in improving access to biomedical information, Methods of Information in Medicine 7 (2) (1992) 1496–1500.

[16] B.L. Humphreys, D.A.B. Lindberg, H.M. Schoolman, G.O. Barnett, The unified medical language system: an informatics research collaboration, JAMIA 5 (1) (1998) 1–11.

[17] M. Kavouras, M. Kokla, Ontology-based fusion of geographic databases, in: FIG Com3 Workshop and Annual Meeting, Athens, Greece, 2000.

[18] R. Krovetz, Homonymy and polysemy in information retrieval, in: 35th meeting of the Association for Computational Linguistics, 1997, pp. 72–79.

[19] D.B. Lenat, R.V. Guha, Building large knowledge-based systems: representation and inference in the CYC project, Addison-Wesley, Reading, MA, 1990.

[20] D.A.B. Lindberg, B.L. Humphreys, A.T. McCray, The unified medical language system, Methods of Information in Medicine 32 (1993) 281–291.

[21] L. Liu, M. Halper, J. Geller, Y. Perl, Controlled vocabularies in OODBs: Modeling issues and implementation, Distributed and Parallel Databases 7 (1) (1999) 37–65.

[22] A.T. McCray, UMLS semantic network, in: Proceedings of the Thirteenth Annual SCAMC, 1989, pp. 503–507.
[23] A.T. McCray, Representing biomedical knowledge in the UMLS semantic network, in: N.C. Broering (Ed.), High-Performance Medical Libraries: Advances in Information Management for the Virtual Era, Meckler, Westport, CT, 1993, pp. 45–55.
[24] A.T. McCray, W.T. Hole, The scope and structure of the first version of the UMLS semantic network, in: Proceedings of the Fourteenth Annual SCAMC, 1990, pp. 126–130.
[25] A.T. McCray, S.J. Nelson, The representation of meaning in the UMLS, Methods of Information in Medicine 34 (1995) 193–201.
[26] D. McDermott, Artificial intelligence meets natural stupidity, in: J. Haugeland (Ed.), Mind Design: Philosophy, Psychology, Artificial Intelligence, MIT Press, Cambridge, MA, 1981, pp. 143–160.
[27] G.A. Miller, WordNet: A lexical database for English, Communications of the ACM 38 (11) (1995) 39–41.
[28] R. Montague, The proper treatment of quantification in ordinary English, in: R. Thomason (Ed.), Formal Philosophy, Yale University Press, New Haven, CT, 1974, pp. 247–270.
[29] Y. Perl, J. Geller, H. Gu, Identifying a forest hierarchy in an OODB specialization hierarchy satisfying disciplined modeling, in: Proceedings of the CoopIS'96, Brussels, Belgium, 1996, pp. 182–195.
[30] M.R. Quillian, Semantic memory, in: M.L. Minsky (Ed.), Semantic Information Processing, The MIT Press, Cambridge, MA, 1968, pp. 227–270.
[31] P.L. Schuyler, W.T. Hole, M.S. Tuttle, D.D. Sherertz, The UMLS Metathesaurus: representing different views of biomedical concepts, Bulletin of the Medical Library Association 81 (2) (1993) 217–222.
[32] S.C. Shapiro, W.J. Rapaport, SNePS considered as a fully intensional propositional semantic network, in: N. Cercone, G. McCalla (Eds.), The Knowledge Frontier, Springer Verlag, NewYork, NJ, 1987, pp. 262–315.
[33] J.F. Sowa (Ed.), Principles of semantic networks, Morgan Kaufmann Publishers Inc., San Mateo, CA, 1991.
[34] J.F. Sowa, Knowledge Representation, Brooks/Cole, Pacific Grove, CA, 2000.
[35] O.N. Suarez-Munist, M.S. Tuttle, N.E. Olson, M.S. Erlbaum, D.D. Sherertz, S.S. Lipow, et al., MEME II supports the cooperative management of terminology, in: J. Cimino (Ed.), Proceedings of the 1996 AMIA Annual Fall Symposium, Washington, DC, October 1996, pp. 84–88.
[36] A. Tarski, Logic, Semantics and Metamathematics, Oxford University Press, Oxford, UK, 1956.
[37] M.S. Tuttle, D.D. Sherertz, M.S. Erlbaum, et al., Adding your terms and relationships to the UMLS Metathesaurus, in: P.D. Clayton (Ed.), Proceedings of the Fifteenth Annual SCAMC, Washington, DC, 1991, pp. 219–223.
[38] M.S. Tuttle, D.D. Sherertz, N.E. Olson, M.S. Erlbaum, W.D. Sperzel, L.F. Fuller, S.J. Nelson, Using meta-1 the first version of the UMLS Metathesaurus, in: Proceedings of the Fourteenth Annual SCAMC, 1990, pp. 131–135.
[39] US Dept. of Health and Human Services, NIH, National Library of Medicine, Unified Medical Language System, 1998.
[40] R. Wille, Restructuring lattice theory: an approach based on hierarchies of concepts, in: I. Rival (Ed.), Ordered Sets, Reidel, Dordrecht, 1982, pp. 445–470.
[41] R. Wille, Concept lattices and conceptual knowledge systems, Computers and Mathematics with Applications 23 (6–9) (1992) 493–515.
[42] W.A. Woods, J.G. Schmolze, The KL-ONE family, in: F. Lehmann (Ed.), Semantic Networks in Artificial Intelligence, Pergamon Press, Oxford, UK, 1992, pp. 133–177.

James Geller received an Electrical Engineering Diploma from the Technical University Vienna, Austria, in 1979. His M.S. degree (1984) and his Ph.D. degree (1988) in Computer Science were received from the State University of New York at Buffalo. He spent the year before his doctoral defense at the Information Sciences Institute (ISI) of USC in Los Angeles, working with their Intelligent Interfaces group. James Geller is currently Professor in the Computer and Science Department of the New Jersey Institute of Technology, where he is also Director of the OODB and AI Laboratory and Vice Chair of the M.S. and Ph.D. programs in Biomedical Informatics. He spent a sabbatical in 1999–2000 at the NEC Research Institute in Princeton, NJ, and is currently a consultant there. Dr. Geller has published numerous journal and conference papers in a number of areas including knowledge representation, parallel artificial intelligence, medical informatics, and object-oriented databases. His current research interests concentrate on object-oriented modeling of medical vocabularies, and on Web mining. James Geller is a past SIGART Treasurer.

Huanying Gu received her B.S. degree in Computer Science from Huazhong University of Science and Technology in 1984 and her M.S. degree in Computer Science from Nanjing University of Science and Technology in 1989, both in China. She received a Ph.D. in Computer Science from the New Jersey Institute of Technology in 1999. Huanying Gu has been with the Health Informatics Department at the University of Medicine and Dentistry of New Jersey as an Assistant Professor since 1999. Her research interests include object-oriented modeling, object-oriented databases, knowledge representation, controlled medical terminologies, and biomedical informatics.

Yehoshua Perl received his Ph.D. degree in Computer Science in 1975 from the Weizmann Institute of Science, Israel. He was appointed lecturer and senior lecturer at the Bar-Ilan University, Israel in 1975 and 1979, respectively. He spent a sabbatical at the University of Illinois in 1977–78. From 1982 to 1985, he was visiting Associate Professor at Rutgers University (New Brunswick). Since 1985, he has been in the Computer Science Department at the New Jersey Institute of Technology (NJIT) where he was appointed as Professor in 1987. He received the Harlan Perlis Research Award of NJIT in 1996. Dr. Perl is the author of more than 80 papers in international journals and conferences. His publications are in object-oriented databases, medical informatics, medical vocabularies, design and analysis of algorithms, design of networks, sorting networks, graph theory, and data compression. From 1995 to 1999, Dr. Perl was involved in the large Object-Oriented Healthcare Vocabulary Repository (OOHVR) project supported by the National Institute of Standards and Technology, ATP Program. Highlights of his research include: The shifting algorithm technique for tree partitioning, analysis of interpolation search, the design of periodic sorting networks, modeling vocabularies using object-oriented databases, and enhancing the semantics of OODBs.

Michael Halper received the B.S. degree (with honors) in computer science from the New Jersey Institute of Technology (NJIT) in 1985; the M.S. degree in computer science from Fairleigh Dickinson University in 1987; and the Ph.D. degree in computer science from NJIT in 1993. During his graduate studies, he was the recipient of a Garden State Graduate Fellowship from the State of New Jersey. Dr. Halper is an Associate Professor of computer science at Kean University, and a visiting researcher at the OODB & AI Laboratory of NJIT. His research interests include conceptual and object-oriented data modeling, part-whole modeling, extensible data models, object-oriented database systems, and medical informatics. He has worked on the OOHVR project—funded by the National Institute of Standards and Technology (NIST) Advanced Technology Program—to model controlled vocabularies and terminologies using object-oriented database technology. Dr. Halper has published numerous papers in international journals, conferences, and workshops. He is a member of The Honor Society of Phi Kappa Phi.