Extending Knowledge Bases through Plausible Inferencing

Michael N. Huhns and Larry M. Stephens
MCC
3500 West Balcones Center Drive
Austin, TX 78759
(512) 338-3651 or huhns@mcc.com

Workshop Session: Knowledge-Base Refinement and Theory Revision

Abstract

This paper considers the composition of tuples from two binary relations in order to derive additional tuples of one of these relations. Our purpose is to determine when the composition is plausible and for which relation the new tuples are derived. The new tuples represent plausible additions to a knowledge base. We first present a formal definition of composition and our extension to it. We next define conditions on the domains and ranges of the relations that are necessary for extended composition to occur. We then show how a set of underlying attributes, independently specified for each relation, is sufficient for determining plausible composition, when the primitives are combined according to an algebra. Finally, we apply our method for extended composition to a representative group of semantic relations and evaluate the results.

1 Introduction

The construction of a large knowledge base is difficult and requires techniques that can facilitate knowledge acquisition. Rather than requiring that all knowledge in the base be entered explicitly, a system could be provided with a basic set of facts and an inference mechanism for inferring additional facts from these [1]. An ideal system would be able to generate all valid inferences, but no invalid inferences. One way to approach this ideal is to provide a set of specialized inference procedures that collectively generate a valid set of inferences. In this paper we develop one such procedure, based on an extended composition of semantic relations from a knowledge base. Figure 1 contains examples of this type of composition. The procedure has the effect of constructing new inference rules, which, when executed, generate extensions to the knowledge base.

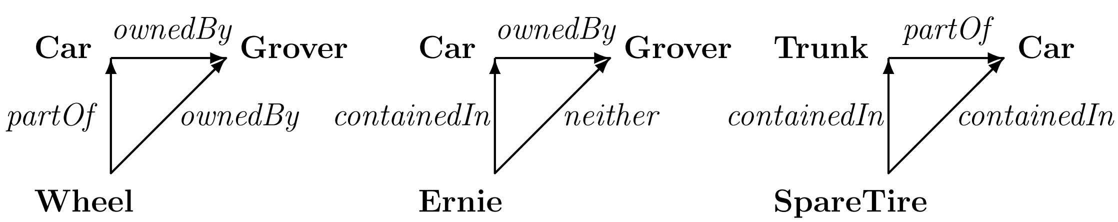


Figure 1: Three examples of the composition of two semantic relations

2 Extended Composition

A binary relation R consists of a set $\mathcal{A}$ (the domain), a set $\mathcal{B}$ (the range), and a mapping that specifies the set of tuples $\langle a, b \rangle$ belonging to R, where $a \in \mathcal{A}$ and $b \in \mathcal{B}$. The mapping may be explicit by listing all the tuples in R or implicit by providing rules for selecting the tuples. In a large frame-based knowledge system, such as CYC [4, 5], the mapping for a relation is only partially specified; other tuples for the relation are added as knowledge is entered. The procedure for composing relations outlined in this paper provides a means of inferring additional tuples belonging to an implicitly defined relation.

A composite relation results from applying the binary operation of *composition* to two binary relations. This operation is defined implicitly by the following definition of a composite relation [7]:

Definition 1 *Let R_i be a relation from set $\mathcal{A}$ to set $\mathcal{B}$ and R_j be a relation from set $\mathcal{B}$ to set $\mathcal{C}$. The* composite relation from $\mathcal{A}$ to $\mathcal{C}$*, denoted $R_i \cdot R_j$, is*

$$R_i \cdot R_j = \{\langle a, c \rangle \mid \wedge \exists b [b \in \mathcal{B} \wedge \langle a, b \rangle \in R_i \wedge \langle b, c \rangle \in R_j]\}$$

We define *extended* composition as follows:

Definition 2 *Let R_i be a relation from set $\mathcal{A}$ to set $\mathcal{B}$ and R_j be a relation from set $\mathcal{C}$ to set $\mathcal{D}$. The* extended composite relation from $\mathcal{A}$ to $\mathcal{D}$*, denoted $R_i \odot R_j$, is*

$$R_i \odot R_j = \{\langle a, d \rangle \mid a \in \mathcal{A} \wedge d \in \mathcal{D} \wedge \exists b [b \in \mathcal{B} \wedge b \in \mathcal{C} \wedge \langle a, b \rangle \in R_i \wedge \langle b, d \rangle \in R_j]\}$$

In the remainder of this paper, we denote $\langle a, b \rangle \in R_i$ by $a.R_i.b$.

If we denote the converse relation of R by R^c, then it can be shown that

$$(R_i \odot R_j \subset R_k) \Leftrightarrow (R_j^c \odot R_i^c \subset R_k^c)$$

Extended composition can also be shown to be associative and not commutative. Extended composition is represented pictorially in Figure 2.

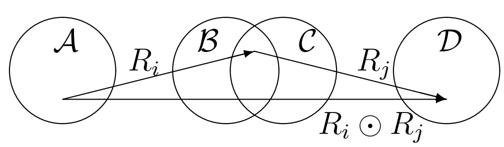


Figure 2: A pictorial representation of the extended composition of relations R_i and R_j

We would like to have an algorithmic way of determining when $R_i \odot R_j$ is nonempty and whether it is a subset of R_i or R_j or neither. Our method for making this determination is based on two premises:

- the domains and ranges of the two relations must be type-compatible, and

- the *primitives* (defined below) of the relations must combine compatibly.

If the first premise is satisfied by relations R_i and R_j, then the primitives of the two relations can be combined to yield the primitives of the composed relation, $R_i \odot R_j$. The primitives of $R_i \odot R_j$ can then be compared to those of R_i and R_j to determine if $R_i \odot R_j$ is a subset of R_i, R_j, both, or neither.

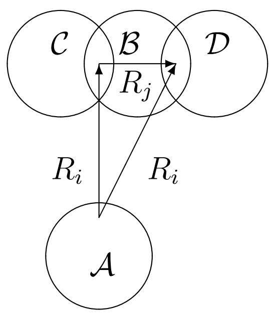
The type compatibility specified by the first premise results in the following necessary conditions for the extended composition of relations:

1. The intersection of sets $\mathcal{B}$ and $\mathcal{C}$ must be nonempty; otherwise, the relation $R_i \odot R_j$ will be empty.

2. For the derived tuples to be elements of R_i, the intersection of sets $\mathcal{B}$ and $\mathcal{D}$ must be nonempty.

3. For the derived tuples to be elements of R_j, the intersection of sets $\mathcal{A}$ and $\mathcal{C}$ must be nonempty.

These conditions, represented using Venn diagrams in Figure 3, eliminate many of the possibilities for extended composition. An algebra based on primitives of the relations eliminates additional implausible compositions.



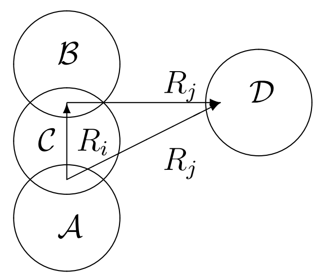


Figure 3: Type requirements on the domains and ranges of R_i and R_j

3 Primitives for Semantic Relations

The second premise above requires a set of primitives that describe each relation and a set of rules for combining primitives. We have postulated a group of ten primitives, based on a literature survey [2, 3, 8, 9] and an analysis of numerous semantic relations in the CYC knowledge base [5]. These primitives are independently determinable for each relation and relatively self-explanatory. They specify a relationship between an element of the domain and an element of the range of the semantic relation being described. The primitives, described next, have values from the set $\mathcal{X} = \{+, 0, -\}$, where + indicates that the relationship holds, − that it does not, and 0 that it is not applicable.

Composable: Some semantic relations can never be meaningfully composed with other relations due to their fundamental characteristics. For example, attributes are not generally transferable through other relations.

Functional: The domain of a Functional relation is in a specific spatial or temporal position with respect to the range of the relation. For example, in an instance of the *componentOf* relation, such as `Wheel`.*componentOf*.`Car`, the `Wheel` is in a specific spatial position with respect to the `Car`. This property does not hold for `Juror`.*memberOf*.`Jury`.

Homeomerous: In each instance of a Homeomerous relation, the element of the domain must be the same kind of thing as the element of the range, e.g., in `PieSlice`.*pieceOf*.`Pie`, the slice is the same stuff as the pie.

Separable: The domain of a Separable relation can be temporally or spatially separated from the range, and can thus exist independently of the range. For the above *componentOf* example, the `Wheel` can be separated from the `Car` and can exist independently. For `Wheel`.*madeOf*.`Aluminum`, the `Aluminum` cannot be separated from the `Wheel` if the `Wheel` is still to exist.

Structural: The domain and range of a Structural relation have a hierarchical relationship in terms of a physical structure. For example, in `Wheel`.*componentOf*.`Car`, the hierarchical structure is from part to whole and the Structural property of *componentOf* has a $-$ value.

Temporal: The domain and range of a Temporal relation are ordered in regard to a temporal structure. For example, there is no notion of time in the relation *pieceOf*, indicated by a value of 0 for Temporal; in *causedBy*, a value of $-$ indicates that the range element precedes the domain element.

Intangible: The domain and range of an Intangible relation have a hierarchical relationship in terms of ownership or mental inclusion. As an example, the relation *ownedBy* has a value of $-$ for Intangible, because the element owned is intangibly included in the owner's sphere of influence.

(Note: values of the last three primitives for the converse of a relation are opposite to those for the forward relation.)

Near: The domain of a relation with property Near is physically or temporally close to the range.

Connected: The domain of a relation with property Connected is physically or temporally connected to the range. A connection, which may be indirect, is indicated by $+$; no connection is denoted by $-$.

Intrinsic: A semantic relation has the property Intrinsic if the relation is an attribute of the stufflike nature of its domain or range. For example, the relation *hasDensity* is an intrinsic property of its domain, so that if `Aluminum`.*hasDensity*.5, then every piece of `Aluminum` inherits this value for its density.

To test our hypotheses, we have selected a representative set of relations, including part-whole, subclass, ownership, causal, and attribution relations. For each of these relations, Table 1 shows the values we have assigned to the above primitives. The domains and ranges of the relations, shown in Table 2, are also needed to determine plausibility.

Table 1: Primitives for Semantic Relations

Relation Name	Relation Primitives									
	Compos.	Func.	Homeo.	Sep.	Struct.	Temp.	Intang.	Near	Conn.	Intrin.
a. componentOf	+	+	−	+	−	0	−	+	+	−
b. memberOf	−	−	−	+	−	0	−	0	−	−
c. pieceOf	+	−	+	+	−	0	−	+	+	+
d. constituentOf	+	−	−	−	−	0	−	+	+	+
e. subeventOf	+	+	−	−	0	−	−	+	+	+
f. subregionOf	+	−	+	−	−	0	−	+	+	+
g. subprocessOf	+	+	+	−	0	−	−	+	+	+
h. subsequenceOf	+	+	+	+	−	−	−	+	+	+
i. purposeOf	+	−	−	+	0	−	0	0	−	−
j. causedBy	+	+	−	+	0	−	0	0	+/−	+
k. producedBy	+	+	−	+	0	−	0	0	−	−
l. ownedBy	+	−	−	+	0	0	−	0	+/−	+
m. focusOf	+	+	−	−	0	0	−	+	+	−
n. connectionOf	+	+	−	+	−	0	−	+	+	−
o. attributeOf	−	−	−	0	0	0	−	0	−	−
p. containedIn	+	−	−	+	−	0	0	+	−	+
r. subfieldOf	+	−	+	−	0	0	−	+	+	+
s. hasMechanisms	+	+	−	+	0	−	0	0	+/−	−
t. isA	+	−	+	+	−	0	−	0	−	+
u. hasWeight	−	−	−	0	0	0	−	0	−	−
v. hasDensity	−	−	−	0	0	0	−	0	−	−

4 Algebra of Relation Primitives

We assume that the results of composing two semantic relations can be determined from the results of combining their ten relation primitives (the accuracy of this assumption is evaluated below) as follows:

$$R_i \odot R_j \equiv V_{R_i} \circ V_{R_j} \tag{1}$$

where $V_R \in \mathcal{X}^{10}$, $\mathcal{X} = \{+, 0, -\}$, and $\circ$ is the combination operator. That is, for the purposes of relation composition, each relation can be represented solely by a vector of values for its ten relation primitives. It thus becomes necessary to define precisely how two of these vectors combine.

We assume that the primitives are orthogonal and form a linear basis for the set of relations. The combination operator $\circ$ can thus be defined in terms of a separate operation table for each primitive, as shown in Table 3. Each operation table is symmetric and has been derived from empirically determined rules for relation composition, such as the following:

- In order to compose, two relations must have the same hierarchical direction for their Structural, Temporal, and Intangible primitives.
- If R_i has the property Connected and R_j does not, then $R_i \odot R_j$ (and $R_j \odot R_i$) cannot have the property Connected. Therefore, $R_i \odot R_j$ (and $R_j \odot R_i$) is not a subset of R_i.
- If R_i has the property Separable and R_j does not, then $R_i \odot R_j$ (and $R_j \odot R_i$) has the property Separable. Therefore, $R_i \odot R_j$ (and $R_j \odot R_i$) may be a subset of R_i.

Table 2: Domains and Ranges for Semantic Relations

Relation Name	Domain	Range
a. componentOf	IndividualObject	IndividualObject
b. memberOf	Thing	Collection
c. pieceOf	Stuff	Stuff
d. constituentOf	Stuff	IndividualObject
e. subeventOf	Event	Event
f. subregionOf	SpatialObject	SpatialObject
g. subprocessOf	Process	Process
h. subsequenceOf	Sequence	Sequence
i. purposeOf	Event	Agent
j. causedBy	Event	Event
k. producedBy	IndividualObject	Process
l. ownedBy	IndividualObject	Tangible&IntangibleObject
m. focusOf	IntangibleObject	IntangibleObject
n. connectionOf	IndividualObject	IndividualObject
o. attributeOf	Attribute	Thing
p. containedIn	IndividualObject	SpatialObject
r. subfieldOf	IntangibleStuff	IntangibleStuff
s. hasMechanisms	Event	Event
t. isA	Collection	Collection
u. hasWeight	TangibleObject	Number
v. hasDensity	TangibleObject	Number

The resultant algebra enables the primitives of the composed relation to be derived. If these derived primitives match the primitives of one (or both) of the composing relations, then a tuple of one (or both) of these can be instantiated; else, the knowledge base can be searched to find all relations that match the resultant primitives, and, if not already instantiated, these can be presented to a user as potential new tuples for the knowledge base.

As an example of this inference procedure, assume that a user has entered the assertions `Wheel`.*componentOf*.`Car` and `Car`.*ownedBy*.`Grover`. Combining the primitives from Table 1 for *componentOf* and *ownedBy* according to the combining rules in Table 3 yields the following vector of primitives for the resultant relation: $V_R = (+ \;\; - \;\; - \;\; + \;\; 0 \; 0 \;\; - \;\; 0 \;\; +/- \;\; +/-)$. This vector matches the primitives of *ownedBy* and does not match those of *componentOf*, thus inferring that `Wheel`.*ownedBy*.`Grover`.

The plausibility of this result is checked by comparing the types of the domain and range of this relation instance with the types specified for *ownedBy* in Table 2. To do this, a taxonomy of types is needed that enables the intersection of domains and ranges to be determined. Such a taxonomy is typically part of frame-based knowledge-representation systems. The types used for our examples are from the CYC ontology [5], a portion of which is reproduced in Figure 4. Using this ontology and Table 2, we find that `Wheel` is an instance of **IndividualObject**, `Grover` is an instance of **Tangible&IntangibleObject**, and these match the domain and range of *ownedBy*. The resultant inference is thus deemed plausible.

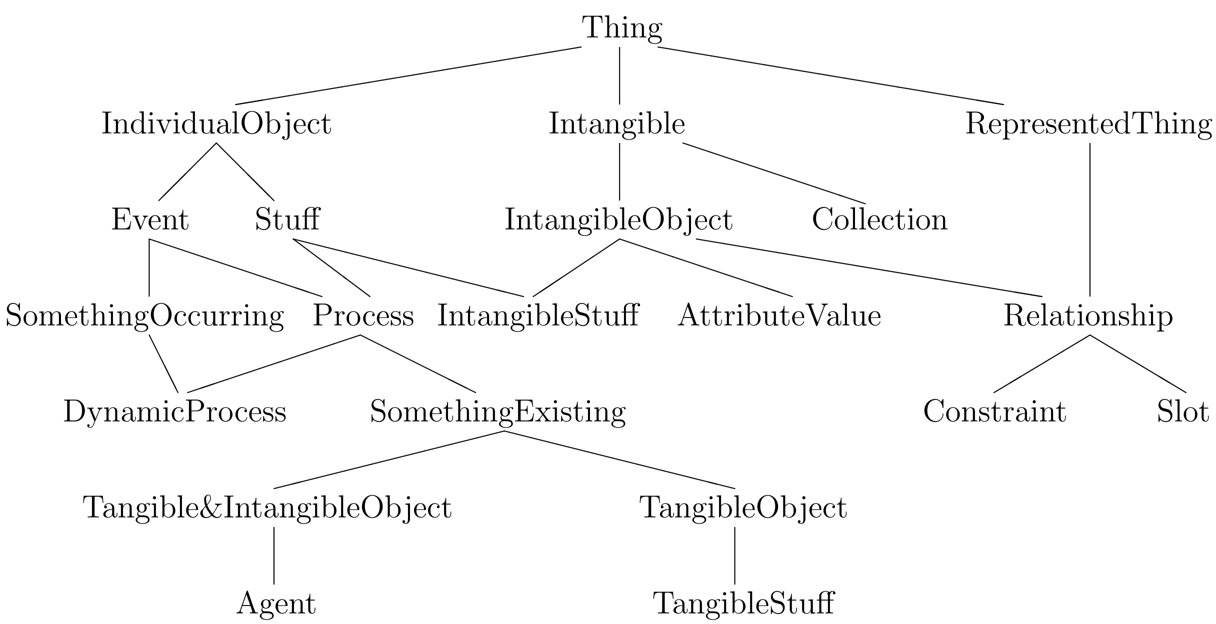


Figure 4: A type hierarchy, taken from the CYC ontology, where each node is a set and each arc denotes a subset relationship

Table 3: Operation Tables for Combining Relation Primitives

Composable

R_i \ R_j	$-$	0	$+$
$-$	P	0	P
0	0	0	0
$+$	P	0	$+$

Functional

R_i \ R_j	$-$	0	$+$
$-$	$+/-$	0	$-$
0	0	0	0
$+$	$-$	0	$+$

Homeomerous

R_i \ R_j	$-$	0	$+$
$-$	$-$	0	$-$
0	0	0	0
$+$	$-$	0	$+$

Separable

R_i \ R_j	$-$	0	$+$
$-$	$-$	0	$+$
0	0	0	0
$+$	$+$	0	$+$

Structural

R_i \ R_j	$-$	0	$+$
$-$	$-$	0	P
0	0	0	0
$+$	P	0	$+$

Temporal

R_i \ R_j	$-$	0	$+$
$-$	$-$	0	P
0	0	0	0
$+$	P	0	$+$

Intangible

R_i \ R_j	$-$	0	$+$
$-$	$-$	0	P
0	0	0	0
$+$	P	0	$+$

Near, Connected

R_i \ R_j	$-$	0	$+$
$-$	$+/-$	0	$-$
0	0	0	0
$+$	$-$	0	$+$

Intrinsic

R_i \ R_j	$-$	0	$+$
$-$	$-$	0	$+/-$
0	0	0	0
$+$	$+/-$	0	$+$

Note: $+/-$ indicates that the relations compose, but that this primitive does not constrain the composition. **P** denotes *prohibited*, indicating that the relations do not compose.

5 Results

The above inference procedure was applied to the set of relations shown in Tables 1 and 2. The results, in the form of a composition matrix, are shown in Table 4. Each entry in Table 4 is equivalent to a rule of the form

$$\forall x \in domain(R_i) \quad \forall y \in [range(R_i) \cap domain(R_j)] \; \forall z \in range(R_j) \quad [x.R_i.y \wedge y.R_j.z \rightarrow x.(R_i \odot R_j).z] \tag{2}$$

The results reflect the order of composition, e.g., $R_j \odot R_i$ as well as $R_i \odot R_j$, which has not been previously addressed [3, 9]. Because each of the operators for combining primitives is symmetric, the composition matrix is nearly symmetric. The only exceptions result from type compatibility, which sometimes excludes a composition from occurring. For example, $f \odot l \subset l$, but $l \odot f = \emptyset$, because the intersection of the range of l with the domain of f is empty.

The following are specific examples of plausible inferences predicted by the extended composition of relations (where $\rightarrow$ denotes logical implication):

- $a \odot l \subset l$
  `Wheel`.*componentOf*.`Car` $\wedge$ `Car`.*ownedBy*.`Grover`
  $\rightarrow$ `Wheel`.*ownedBy*.`Grover`

- $a \odot p \subset p$
  `Tire`.*componentOf*.`Car` $\wedge$ `Car`.*containedIn*.`Garage`
  $\rightarrow$ `Tire`.*containedIn*.`Garage`

Table 4: Composition Matrix for $R_i \odot R_j$

R_i	**R_j**																				
	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	r	s	t	u	v
a	a	-	-	-	-	-	-	a	-	-	-	l	-	a	-	p	-	-	-	-	-
b	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
c	-	-	c	-	-	c	-	c	-	-	-	l	-	-	-	p	-	-	-	-	-
d	-	-	-	d	-	d	-	-	-	-	-	l	-	-	-	p	-	-	-	-	-
e	-	-	-	-	e	-	e	-	i	j	k	l	m	-	-	-	-	s	-	-	-
f	-	-	c	d	-	f	-	-	-	-	-	l	-	-	-	p	-	-	-	-	-
g	-	-	-	-	e	-	g	-	i	j	k	l	m	-	-	-	r	s	-	-	-
h	a	-	c	-	-	-	-	h	i	j	k	l	-	n	-	p	-	-	-	-	-
i	-	-	-	-	i	-	i	i	i	i	i	-	-	-	-	-	-	i	-	-	-
j	-	-	-	-	j	-	j	j	i	j	jk	-	-	-	-	-	-	js	-	-	-
k	-	-	-	-	k	-	k	k	i	jk	k	-	-	-	-	-	-	ks	-	-	-
l	l	-	l	l	l	-	l	l	-	-	-	l	-	l	-	-	-	-	-	-	-
m	-	-	-	-	m	-	m	-	-	-	-	l	m	-	-	-	-	-	-	-	-
n	a	-	-	-	-	-	-	n	-	-	-	l	-	n	-	p	-	-	-	-	-
o	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
p	p	-	p	p	-	p	-	p	-	-	-	-	-	p	-	p	-	-	-	-	-
r	-	-	-	-	-	-	r	-	-	-	-	l	-	-	-	-	r	-	-	-	-
s	-	-	-	-	s	-	s	-	i	js	ks	-	-	-	-	-	-	s	-	-	-
t	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	t	-	-
u	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
v	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Note: the letters in this matrix refer to the relations listed in Tables 1 and 2.

- $p \odot a \subset p$
  `Refrigerator`.*containedIn*.`Kitchen` $\wedge$ `Kitchen`.*componentOf*.`House`
  $\rightarrow$ `Refrigerator`.*containedIn*.`House`

- $d \odot f \subset d$
  `Silicon`.*constituentOf*.`Beach` $\wedge$ `Beach`.*subregionOf*.`Island`
  $\rightarrow$ `Silicon`.*constituentOf*.`Island`

- $i \odot e \subset i$
  `Thunder`.*causedBy*.`Lightning` $\wedge$ `Lightning`.*subeventOf*.`ThunderStorm`
  $\rightarrow$ `Thunder`.*causedBy*.`ThunderStorm`

- $g \odot m \subset m$
  `Chewing`.*subprocessOf*.`Eating` $\wedge$ `Eating`.*mechanismOf*.`DigestingFood`
  $\rightarrow$ `Chewing`.*mechanismOf*.`DigestingFood`

The technique for relation composition also correctly predicts when neither of the composed relations can be inferred. For example

- $p \odot l = \emptyset$
  `Grover`.*containedIn*.`Car` $\wedge$ `Car`.*ownedBy*.`Ernie`
  $\not\rightarrow$ (`Grover`.*containedIn*.`Ernie` $\vee$ `Grover`.*ownedBy*.`Ernie`).

6 Discussion and Conclusions

The inference procedure and results presented in this paper extend the work of [2, 3, 8, 9]. Chaffin and Hermann [2] identify a set of *relation elements* (relation primitives) that can be used to describe and classify relations. Each relation element is a fundamental property that holds between the domain and range of the relation.

Winston, Chaffin, and Herrmann [9] define three independent relation elements, *inclusion*, *connection*, and *similarity*; these are used to describe spatial inclusion, meronymic inclusion, and class inclusion. When any inclusion relation is combined with another, they find that a valid inference can be made and that the resultant relation is the one having the fewest relation elements. In addition, Winston *et al.* identify three dependent elements of *connection* that explain the transitivity, but not the composability, of six meronymic relations.

Cohen and Loiselle [3] identify two deep structures for relations: *hierarchical* and *temporal*, each having a direction. Each relation is hierarchical, temporal, or both. When two relations are composed, the resultant relation may have any of several possible deep structures, depending on the properties of the composing relations. They found that inferences are most plausible when either the hierarchical or temporal directions of the two composing relations are the same as that in the composed relation. Like Winston *et al.*, they do not consider type consistency in composing relations.

We extend the research efforts cited above by basing relation composition on set theory. On this basis, we conclude that typing of the domain and range elements may restrict composition, independently of any relation attribute restrictions. In addition, we extend the work of [9] by explicitly considering the hierarchical nature of the inclusion relations, as suggested by [3]. This leads to a means of defining the primitive attributes of the *converse* of a relation and, consequently, of composing a converse with other relations.

We provide a vector of ten primitives for each of 21 typical relations. This vector representation provides a more powerful basis for ranking and classifying relations than does the linear ordering in [9]. Since there are three possible values for each of the ten primitives, our representation provides for $3^{10} = 59,049$ different basis vectors that can be used to represent relations. The number of relations that could be represented is actually much greater because of the large number of types that could be chosen for the domains and ranges.

The inference procedure we developed for relation composition is based on several assumptions. The foremost of these is that relation composition is equivalent to a combination of the corresponding vector of primitives. The correctness of this assumption is borne out by the plausibility of the predicted inferences, shown in Table 4. A second assumption is that each relation primitive is orthogonal to the others. This simplifying assumption greatly increases the efficiency of the inference procedure by yielding operation tables (see Table 3) that are independent of each other. Although the validity of the results supports this assumption also, there is some evidence that the chosen primitives are NOT orthogonal. For example, the primitives Connected, Homeomerous, and Intrinsic combine dependently according to the following rule to yield compositions with attribute relations not predicted by our algebra:

$$(\textit{attributeOf}.\text{Intrinsic}.+) \wedge (R_j.\text{Connected}.+) \wedge (R_j.\text{Homeomerous}.+) \\ \rightarrow (\textit{attributeOf} \odot R_j \subset \textit{attributeOf})$$

Such a rule would yield the valid inference *densityOf* $\odot$ *pieceOf* $\subset$ *densityOf*, which does not result from our relation algebra. It could be applied after extended composition and viewed as an additional inference mechanism.

Other valid inferences are missing from Table 4, including *memberOf* $\odot$ *isA* $\subset$ *memberOf* and *componentOf* $\odot$ *attributeOf* $\subset$ *attributeOf*. However, we feel that these omissions do not diminish the utility of our results, in that our procedure is designed for correctness instead of completeness. In addition, many knowledge-based systems have other inference mechanisms that could generate these missing inferences. For example, an automatic classifier [6] would generate the inference *memberOf* $\odot$ *isA* $\subset$ *memberOf*.

The potential for generating new inferences in a large knowledge base, such as the one in CYC, is enormous. CYC, currently with >4000 relations, could have approximately eight million possible compositions. Of these, 20% are predicted to be plausible, based on the percentage of valid entries in Table 4. For all possible values of relation primitives, no more than 31% could be composed validly due to *prohibited* entries in the operation tables for combining primitives. The 30,000 assertions now in the CYC knowledge base can be combined using the predicted compositions to yield many new inferences.

However, there are two major problems with extended composition. First, reason maintenance for the resultant inferences is computationally problematic, because the inferences depend not only on the relations being composed, but also on the relation primitives for all of the relations involved. Second, assigning values for the relation primitives is conceptually problematic. The values are subjective and must be entered manually for each relation in a knowledge base. The validity of the inferences generated by extended composition are directly dependent on these values.

Nevertheless, we expect that the relation primitives can be used for classifying relations, as well as generating new inferences, and for suggesting plausible analogies. The procedure for extended composition appears to be a viable technique for increasing the information in an existing knowledge base. Because the procedure has the effect of generating new inference rules and then applying them, it yields plausible inferences that are not within the deductive closure of the original knowledge base.

References

[1] M. Baker, M. H. Burstein, and A. M. Collins, "Implementing a Model of Human Plausible Reasoning," *Proceedings of the Tenth IJCAI*, Milan, Italy, August 1987, pp. 185–188.

[2] R. Chaffin and D. J. Herrmann, "Relation Element Theory: A New Account of the Representation and Processing of Semantic Relations," in D. Gorfein and R. Hoffman, eds., *Memory and Learning: The Ebbinghaus Centennial Conference*, Lawrence Erlbaum, Hillsdale, NJ, 1987.

[3] P. R. Cohen and C. L. Loiselle, "Beyond ISA: Structures for Plausible Inference in Semantic Networks," *Proceedings AAAI88*, St. Paul, MN, August 1988, pp. 415–420.

[4] D. B. Lenat, M. Prakash, and M. Shepherd, "CYC: Using Common Sense Knowledge to Overcome Brittleness and Knowledge Acquisition Bottlenecks," *AI Magazine*, vol. 6, no. 4, Winter 1986, pp. 65–85.

[5] D. Lenat and R. V. Guha, "The World According to CYC," MCC Technical Report No. ACA-AI-300-88, MCC, Austin, TX, September 1988.

[6] T. Lipkis, "A KL-ONE Classifier," *Proceedings of the 1981 KL-ONE Workshop*, J. G. Schmolze and R. J. Brachman, eds., BBN Report No. 4842, Bolt Beranek and Newman Inc., June 1982, pp. 128–145.

[7] D. F. Stanat and D. F. McAllister, *Discrete Mathematics in Computer Science*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1977.

[8] A. Wierzbicka, "apples are not a 'kind of fruit': the semantics of human categorization," *American Ethnologist*, vol. 11, 1984, pp. 313–328.

[9] M. E. Winston, R. Chaffin, and D. Herrmann, "A Taxonomy of Part-Whole Relations," *Cognitive Science*, vol. 11, 1987, pp. 417–444.