THE ROLE OF LITERAL MEANING IN THE COMPREHENSION OF NON-LITERAL CONSTRUCTIONS

STEVEN L. LYTINEN, ROBERT R. BURRIDGE, AND JEFFREY D. KIRTNER

Artificial Intelligence Laboratory, University of Michigan, Ann Arbor, MI 48109

Based on psychological studies which show that metaphors and other non-literal constructions are comprehended in the same amount of time as comparable literal constructions, some researchers have concluded that literal meaning is not computed during comprehension of non-literal constructions. In this paper, we suggest that the empirical evidence does not rule out the possibility that literal meaning is constructed. We present a computational model of comprehension of non-literal expressions which is consistent with the data, but in which literal meaning is computed. This model has been implemented as part of a unification-based natural language processing system, called LINK.

Key words; natural language processing, metaphor, metonymy, unification grammars.

1. INTRODUCTION

Many theories of the semantic interpretation of natural languages assert that semantic interpretations are *compositionally* constructed; that is, the interpretation of an utterance is a fairly straightforward combination of the interpretations of its constituents. Katz and Fodor (1963) define compositionality as follows:

> On the basis of his knowledge of the grammatical properties and the meanings of the morphemes of the language, the rules the speaker knows enable him to determine the meaning of a novel sentence in terms of the manner in which the parts of the sentence are composed to form the whole. (Katz and Fodor 1963; p. 175)

Problematic for compositional theories are so-called "non-literal" constructions such as metaphors, metonymies, figures of speech, idioms, and indirect speech acts. They are problematic because there is no straightforward way to construct their intended interpretations from combining the meanings of the morphemes that they are composed of. For example, consider the metaphor "No man is an island." Literally, the word "island" refers to a body of land surrounded by water, but this sentence has an intended meaning beyond the obvious statement of fact that male humans are not bodies of land surrounded by water.

How, then, are metaphors and other non-literal constructions understood? A point of contention with regard to this question has been whether or not it is necessary to compute the literal meaning of a non-literal utterance en route to understanding its intended (non-literal) meaning. Many theories developed in both AI and psychology have extended the standard compositional paradigm to account for comprehension of non-literal constructions by asserting that literal meaning is constructed by standard compositional means, and then literal meaning is transformed to intended meaning by a separate set of rules. However, a large body of research in psycholinguistics has revealed that, in an appropriate context, metaphors and other non-literal constructions are comprehended no slower than comparable literal constructions, and sometimes even faster. This evidence, along with other related studies, has led others to conclude that computation of literal meaning is not a necessary step in the understanding of metaphors.

In this paper, we suggest that the existing psychological data does not provide the necessary evidence to determine whether or not literal meaning is computed during the

comprehension of metaphors and other non-literal expressions. Moreover, from a computational standpoint, there are reasons to prefer an approach in which literal meaning is always computed. We present such an approach, and show that this model of metaphor processing is also consistent with the existing psychological data.

An important feature of our model which distinguishes it from standard extensions of the compositional paradigm is that the non-literal interpretation of a potentially non-literal expression is *automatically* computed from the expression's literal meaning, regardless of the context in which it appears. We argue that the existing data cannot distinguish between this processing model and one in which literal meaning is not computed.

Our approach to processing non-literal expressions has been implemented in a natural language processing (NLP) system called LINK. LINK is an integrated, unification-based system, in which syntactic, semantic, and domain knowledge are all implemented in unification *constraint rules*. In order to process non-literal constructions, we have added a set of *mapping rules,* which construct alternative (non-literal) interpretations from literal meanings constructed during parsing. Context then determines which of the possible interpretations of the expression is most appropriate.

This paper is organized as follows: first, we present an argument for why, from a computational standpoint, there is a reason for computing the literal meaning of a metaphor and for the use of mapping rules. Next, we examine previous computational models which have proposed the use of mapping rules, and the psychological evidence which seems to refute these theories. Finally, we present our model, and show why it is consistent with the psychological data.

2. THE NEED FOR MAPPING RULES

Let us explore further why non-literal constructions are problematic for compositional theories of semantics. Consider the following metaphor:

The stock market rose today.

As we discussed above, essential to any compositional approach is the notion that the meaning of a sentence can be constructed from the meanings of its pieces (either lexical items or phrases). However, if we combine the literal meanings of "stock market" and "rose," we obviously will not arrive at the correct interpretation of this sentence. We could define one or both of these terms as ambiguous, with "literal" and "metaphoric" senses, but it is not clear that we can easily separate the meaning of the metaphor into components to be associated with individual lexical items, as the compositional approach requires. For example, "rose" could be defined ambiguously, as either referring to an increase in altitude or an increase in a numeric indicator, but this does not capture the generalization that almost any word which refers to a change in altitude can be used in the same way. Consider the following examples:

The stock market plummeted today.
My fever has gone up.
Computer science enrollments are leveling off.
The 10% jump in property taxes this year was outrageous.

If we defined "rose" as ambiguous, we would have to define every other word which referred to a change in altitude ambiguously, also. This is inconvenient at best, and implausible at worst.

The same problem arises with other types of non-literal constructions, such as metonymies, in which an object is referred to by mention of a related object. For example:

John loves to read Shakespeare.

Obviously, this sentence means that John loves to read the works of Shakespeare, not the author himself. We could encode the word "Shakespeare" as ambiguous, but as with our metaphor example, this metonymy is quite productive; that is, many similar examples can be found which use different lexical items and syntactic constructions:

I am listening to Beethoven.
This painting is a genuine Picasso.
Millions of people watched CNN during the Gulf War.

Once again, it seems that the compositional approach fails to capture the generality of this construction: if we simply define words like "Shakespeare" and "Beethoven" as having multiple meanings, we have to define similar multiple meanings for a very large number of other lexical items.

Examples like these are not uncommon. We will not attempt to argue this point here, as many others have presented convincing arguments as to the prevalence of highly productive metaphors and other non-literal constructions (e.g., Lakoff and Johnson 1980; Langacker 1987). Given their prevalence, it is important that we properly account for them, with a mechanism which explains their productivity.

How can we write appropriately general interpretation rules for metaphors, metonymies, and so on? Since they are often syntactically and lexically flexible, it seems that the only way to do so is at the semantic level. Instances of a given metaphor or metonymy share commonalities in "literal" meaning (e.g., all lexical items that can be used in the "rose" metaphor normally refer to a change in altitude) and also in "intended" meaning (e.g., when used metaphorically, words like "rose" refer to a change in value of a numerical indicator). Thus, it seems natural to write an interpretation rule that transforms the literal meaning of a construction to an alternate non-literal one.

We will refer to this kind of rule as a *mapping rule*. In the case of the "rose" metaphor, the mapping is from a change in altitude to a change in value of a numerical indicator. The direction of the change in value is "the same" as the direction of the change in altitude: that is, increase in altitude means increase in the numerical indicator. In the case of our metonymy example, the mapping is from the producer of an item (Shakespeare, Beethoven, CNN) to the product (literature, music, news coverage).

How can mapping rules be used in semantic interpretation? Comprehension of a non-literal construction using a mapping rule would involve constructing the literal interpretation of the construction, identifying the appropriate mapping rule, and applying the rule to the literal interpretation, thereby yielding the intended (non-literal) interpretation.

In addition to capturing the productivity of non-literal constructions, mapping rules can also account for the ease with which people can understand many novel uses of a construction. This ease of comprehension is difficult to explain in the standard compositional paradigm. For example, a few years ago, before the current recession began, as economic growth slowed, economists started talking about a "soft landing" of the economy. This conveys a rather complex process of growth slowly decelerating until it levels off at around zero (little or no growth), without slipping into recession (negative growth). Despite the complexity of this metaphor, when the term was introduced into the American vocabulary, it seemed to require no explanation. A theory of metaphor comprehension based on lexical ambiguity cannot easily account for this, since "soft landing" previously had no metaphorical definition.

3. THE STANDARD PRAGMATIC MODEL

Many previous comprehension theories of non-literal expressions have proposed the use of mapping rules (e.g., Searle 1978; Carbonell 1982; Induryka 1987). In fact, this approach has been called the Standard Pragmatic Model (Gerrig 1989). Gerrig outlined three general steps in the Standard Model:

1. Consruct the literal meaning of the utterance.
2. Identify a failed semantic or pragmatic constraint in the literal meaning.
3. Apply a mapping rule to transform the literal meaning to the intended meaning.

Variants of this model have suggested more general reasoning or search techniques, rather than mapping rules, for constructing intended meaning once a failed constraint has been identified (e.g., Fass 1988; Ballim, Wilks, and Barnden 1991).

The Standard Pragmatic model makes two predictions about processing. First, since steps 2 and 3 must be performed after completion of step 1, it predicts that comprehension of metaphors is slower than comprehension of comparable literal utterances. Second, since step 3 above is only executed if "normal" semantic and pragmatic processing fails to interpret an utterance, it predicts that the metaphorical interpretation of an utterance is not computed in a context in which the literal interpretation is acceptable.

Numerous psychological studies contradict both of these predictions. In general, these studies have used two methodologies to examine how non-literal constructions are understood. One technique is to compare reading times for non-literal and comparable literal expressions. The comparison suggests what processes are present or absent in the comprehension of non-literal expressions. A second technique is to look for evidence of the activation of literal and non-literal meanings of expressions in various contexts. This provides evidence for whether or not literal and non-literal meanings have been computed during comprehension. We will discuss both of these classes of results.

3.1. Reading Times

Many studies have shown that, in an appropriate context, metaphor comprehension is as fast as comprehension of comparable literal expressions. Ortony et al. (1978) reported two reaction time experiments designed to test how long it takes subjects to understand literal and metaphoric sentences in context. The method used was as follows: subjects read one or more sentences which set up a context, then read the target sentence, and then indicated as quickly as possible that they understood the target by pressing a key. In experiment 1 the context was either short or long. In the short context, subjects took longer to understand non-literal than literal targets, but in long contexts the reading times were not significantly different. In experiment 2, the targets were phrases that could have either a literal or idiomatic intepretation, depending on context. It was found that targets with an idiomatic interpretation took no longer to process than the literal interpretations of the same targets.

Gibbs (1979) reported similar results for indirect requests. Subjects read stories, one line at a time, which ended in either an indirect request (e.g., "Must you open the window?") or a direct request (e.g., "Please open the window"). In an appropriate context, the indirect requests took no longer to comprehend than the direct requests.

Gildea and Glucksberg (1983) addressed the question of what constitutes an appropriate context for facilitation of metaphor processing. The authors proposed that any context which activates a property of the metaphor vehicle (the predicate in "All X are Y" sentences), and which is also informative about the topic of the metaphor (the subjects in

"All X are Y" sentences) will be sufficient to trigger immediate comprehension. That is, contexts which activate important properties of the metaphor ground (the concept underlying the metaphor) facilitate immediate, automatic metaphor interpretation.

The authors tested this proposal by examining the relative effectiveness of three kinds of priming contexts: activation of the figurative sense of the metaphor ground; activation of the literal sense of the metaphor ground; and activation of the general semantic field of the ground. All three types of contexts led to immediate and automatic metaphor comprehension. The proof of immediate comprehension lies in the metaphor interference effect, as reported in Glucksberg et al. (1982) below. The results of these experiments showed that any context, literal or figurative, that activates the ground concept of a metaphor will facilitate metaphor comprehension equally well.

Inhoff et al. (1984) conducted several experiments to discover more about how context affects the processing of both literal and non-literal sentences. The first two experiments addressed two issues: First, do metaphors really take longer to process given a short context, as found by Ortony et al., or was it simply that in Ortony's experiment the short contexts turned out to be bad contexts? A related question is: how long do literal sentences take to process given a bad context in which to interpret them? Second, given longer contexts, are metaphors and literal sentences really processed in the same amount of time?

In the first experiment, Ortony's results were replicated. Short contexts increased metaphor processing times, but not literal processing times. In the second experiment, Ortony's results were again replicated. Long contexts facilitated metaphor processing and literal processing equally.

Inhoff reports a third experiment that addressed two further issues: First, will a short context (only one sentence) which is designed to activate the appropriate literal or figurative meaning of the target result in equal processing times for both metaphor and literal targets? Second, is there any support for the "process priming" hypothesis? This hypothesis asserts that if there is a separate "metaphor processing unit" that is activated during the processing of metaphors but not during literal language, then the processing of a figurative context should speed up the processing of a figurative target. The process priming hypothesis predicts that figurative primes will better facilitate target understanding than literal target understanding (because the "metaphor processing unit" has already been activated by processing the context); and conversely, literal primes will facilitate literal targets better than figurative targets.

The results from experiment 3 were as follows: first, there was no evidence the metaphoric targets took longer to process than literal targets, even though all contexts were short. That is, the *quality* of context is what counts, not its length; second, there was some support for the process priming hypothesis, because figurative contexts better facilitated figurative targets, and literal contexts better facilitated literal targets. However, these results only applied within each trial; there was no similar priming effect between trials; i.e., a metaphoric target in trial J did not prime a metaphoric context/target in trial J + 1. The evidence for the process priming hypothesis is therefore inconclusive.

More recent experiments have on the whole confirmed that many types of non-literal statements take no longer to process than literal statements. See Gerrig (1989) or Gibbs (1984) for a more extensive discussion of the evidence.

3.2. Activation of Literal and Non-literal Meanings

Gibbs has also examined the question of whether or not literal meanings of indirect requests are activated during comprehension. In (Gibbs 1983), he found no evidence for activation of literal meaning if an indirect request was comprehended in an appropriate

context. Immediately after reading an indirect request such as "Can you pass the salt?" subjects were asked to make sentence/nonsentence judgments on target sentences. When the target was a paraphrase of the literal meaning of the indirect request (e.g., a target about ability to pass the salt), there was no facilitation of this judgment task from the indirect request preceding it. However, in the reverse situation, facilitation did occur: subjects were faster at making sentence/nonsentence judgments on paraphrases of the indirect request meaning if the request appeared in a context in which it was interpreted literally. Gibbs concluded from this assymmetry of priming that literal meaning of indirect requests was not computed, but that the non-literal meaning always was.

Cacciari and Tabossi (1988) also found that, for some idioms, literal meaning is not activated after the final word of an idiomatic construction. Subjects listened to sentences which contained idioms, such as "After winning the match, the tennis player was in seventh heaven." Immediately after hearing the final word of the idiomatic construction (in this case, "heaven"), subjects were given a visual word judgment task. Words which were semantically related to the idiomatic meaning were primed, but targets that were related to the literal meaning of the final word in the idiom were not. However, Cacciari and Tabossi found that this result depended on the familiarity of the idioms used. In a follow-up experiment, subjects were pretested to see if they could complete an idiom when shown the first few words in the idiom. Subjects who succeeded in this completion task were screened, and a similar priming experiment was conducted. This time, it was found that immediately after hearing the final word of an idiom, words related to the literal meaning of the final word were primed, but words related to the meaning of the idiom were not. After 300 msecs, targets related to the idiomatic meaning became primed also.

Glucksberg et al. (1982) also addressed the question of when literal and non-literal meanings are computed, this time for metaphors. Subjects were asked to make judgments about the literal truth of sentences of the form "All/Some X are Y." Subjects took significantly longer to judge the truth of sentences which had a reasonable metaphoric meaning than those which did not. That is, judgment about the literal truth of sentences such as "All men are wolves" was slower than for "All men are telephones." Glucksberg et al. called this the *metaphor inference effect*. According to their analysis, the presence of the metaphor interfered with subjects' ability to judge sentences literally false. Since the task only required subjects to make judgments based on literal meaning, the authors claimed that this interference was evidence that subjects computed the metaphoric meaning anyway. This suggests that metaphor comprehension is an automatic process, which contradicts the Standard Model.

3.3. Interpreting the Psychological Data

The evidence against the Standard Theory has led some to conclude that literal meaning is not computed during comprehension of non-literal constructions. Perhaps the strongest proponent of this view is Gibbs, who goes so far as to question the validity of the notion of literal meaning:

> Sentences do not have well-defined literal meanings, regardless of whether these are determined in light of a set of background assumptions. Moreover, the putative literal meanings of sentences do not contribute in systematic ways toward the understanding of speakers' utterance meanings. These observations suggest that the distinctions between literal and metaphoric meanings . . . have little psychological validity. (Gibbs 1984; p. 275)

The conclusion seems to be that, since the Standard Model makes incorrect predictions, all three main steps of the model are incorrect. However, it is not clear that this

conclusion necessarily follows from the data. Let us examine this question more closely. Summarizing the experiments discussed above, we must explain three findings:

1. Reading times for non-literal constructions are no slower than reading times for literal constructions.
2. Non-literal meanings appear to be computed even when expressions are intended to be taken literally.
3. Literal meanings are not primed after comprehension of a sentence containing a non-literal construction.

Do these three findings necessarily imply that literal meaning is not computed during comprehension? The first two findings are problematic for the Standard Model not because it states that literal meaning is computed first, but rather because of the failure-driven nature of non-literal processing. A non-literal interpretation is not constructed unless some difficulty is encountered in constructing a literal interpretation. Waiting for a semantic (or pragmatic) constraint violation before invoking a mapping rule results in the incorrect prediction of additional processing time for metaphors, as well as the incorrect prediction that non-literal meanings will not be computed during comprehension of literal utterances. Thus, it seems to be step 2 of the Standard Pragmatic Model that results in the incorrect predictions.

The third finding, that literal meanings are not primed after comprehension of a non-literal sentence, also does not imply that literal meanings are *never* constructed during comprehension. It is entirely possible that literal meanings have already been activated and then suppressed by the end of a sentence. We will have more to say about this in Sects. 5 and 6.

4. INTERPRETATION OF NON-LITERAL CONSTRUCTIONS IN LINK

We now present our alternative theory of semantic interpretation, which is implemented in the LINK system. Though our model is similar to the Standard Pragmatic Model in that it utilizes mapping rules, it differs from the Standard Model in the way in which these rules are applied.

In LINK, processing follows these general steps:

1. Incrementally construct the (literal) interpretations of constituents in the utterance.
2. As interpretations are constructed, try to match them against mapping rules. If any rules match, also construct the result of the mapping (the non-literal interpretation) as an *alternative* interpretation of the constituent.
3. Use contextual information to determine which of the candidate interpretations is (are) acceptable.

This model is a refinement of Martin's (1990) theory of metaphor processing.[1] It differs from Martin's approach in three ways: first, the construction of non-literal interpretations in LINK is an incremental process, which proceeds during the comprehension of a sentence. In Martin's system, all metaphoric interpretations were constructed after completion of parsing. Second, mapping rules in LINK are used to process other non-literal constructions, such as metonymies and figures of speech. Martin's rules were used primarily in metaphor comprehension. Finally, because the application of mapping rules is integrated

---

[1] In Martin's theory, the initial representation constructed was called the "primal representation" of the utterance. We view this as equivalent to literal meaning, although it is not at all clear that Martin would agree.
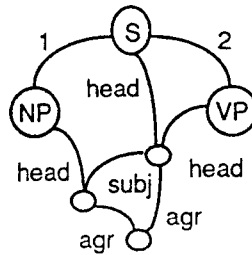
FIGURE 1.   DAG version of the S rule.

with parsing, LINK's mapping rules can contain a mixture of syntactic and semantic information. Martin's rules operated strictly in the domain of semantics. The ability to include other kinds of information in LINK's mapping rules is important in the processing of other types of non-literal constructions.

Before we can explain the details of LINK's mapping rules, we must give a general overview of LINK.

4.1.   LINK's Unification Grammar

All syntactic, semantic, and pragmatic knowledge is encoded in LINK in unification *constraint rules*. These rules are very similar in form to other unification grammars, such as PATR-II (Shieber 1986) or HPSG (Pollard and Sag 1987). Each constraint rule consists of a set of equations, each of which constrains the interpretation which the parser can build in some way, by limiting, for a class of nodes (i.e., any node with a particular label), the set of arcs that can lead from a node of that class, as well as the types of nodes that arcs can lead to.

Here is a simplified example of a constraint rule:

```
(define-class S
      (1)  =  NP                                <1>
      (2)  =  VP                                <2>
      (head)  =  (2 head)                       <3>
      (head agr)  =  (1 head agr)               <4>
      (head subj)  =  (1 head))                 <5>
```

Each equation in this rule specifies a property which says any node labeled S must have. A property consists of a *path,* or a sequence of arcs with the appropriate labels starting from the node in question; and a *value,* which is another node to be found at the end of the path. Equations specify the values of properties in one of two ways. They may specify the label of the node to be found at the end of the path, as in equations 1 and 2 (i.e., the arc from an S node labeled 1 leads to a node labeled NP). We will call these *labeling equations.* Or, they may specify that two paths must lead to the identical node, as in Eqs. (3–5). Identity here is defined by the *unification* operation; i.e., if two paths must lead to the identical node, then the nodes at the end of the two paths must unify. Unification merges the properties of two nodes; thus, two paths can unify if their values have no properties which explicitly contradict each other. These equations will be called *unifying equations.*

Constraint rules can be represented as directed acyclic graphs (DAGs), whose nodes and links are labeled. The DAG form of the above rule is given in Fig. 1. The root node

of this DAG is labled S, because that is the class of constituent to which this rule applies. For each property specified in the rule, there is a corresponding path in the DAG leading from the root node to a node with the appropriate label (in the case of labeling equations), or there are two paths from the root node which lead to the same node (in the case of unifying equations). Thus, in Fig. 1, a node labeled *NP* is found at the end of the arc labeled 1 coming from the S node.

Functionally, the above rule encodes information about English sentences as follows. Equations (1) and (2) specify that a sentence is made up of two subconstituents: an NP and a VP. Ordering of these constituents is implicit in the numbering of the paths. Equation 3 assigns the HEAD of the sentence to be the VP, by unifying the VP's HEAD path with the HEAD path of the S. This will be discussed further shortly. Equation (4) specifies that the NP and the VP must agree in number and person. These syntactic properties are found under the AGR (agreement) feature of each constituent. Finally, Equation (5) assigns the NP to be the subject of the sentence.

The HEAD property referred to in Eqs. (3)–(5) is used to propagate information up and down the DAG. This is accomplished by unification of HEAD links, as in Eq. (3). Because of this equation, any information on the HEAD of the VP is accessible from the S node. Similar equations would unify the heads of other constituents, such as a verb (V) and a VP, or a particular lexical item and a V.

Lexical items typically provide the values which are propagated by HEAD links. They are encoded in the same form as grammar rules. For example:

```
(define-class V
    (1) = eats                                    <6>
    (head agr number) = sing                      <7>
    (head agr person) = 3rd                       <8>
    (head rep) = EAT-FOOD                         <9>
    (head subj rep) = (head rep actor)           <10>
    (head dobj rep) = (head rep object))         <11>
```

Typical values provided by lexical rules include syntactic feature information, such as the AGR feature; as well as semantic information, which causes a semantic interpretation of the sentence to be constructed as parsing proceeds. In this particular lexical entry, Equations (7)–(8) specify the word's syntactic features, found under the AGR property. Equation (9) provides semantic information about the word, specifying that "eats" means EAT-FOOD. Equations (10) and (11) specify mappings from syntactic to semantic dependencies. Equation (10) states that whatever constituent fills the SUBJECT role will also be assigned as the ACTOR of the EAT-FOOD. Similarly, 11 specifies that the syntactic direct object (DOBJ) is assigned as the semantic OBJECT.

Equations (10) and (11) are used in conjunction with the system's domain knowledge, to impose restrictions on the semantic properties (i.e., the values of the REP path) of the subject and direct object of "eats" (i.e., the ACTOR and OBJECT of EAT-FOOD). Domain knowledge is also encoded in constraint rules. In this particular case, the relevant rule is the following:

```
(define-class EAT-FOOD
    (actor) = HUMAN                <12>
    (object) = FOOD                <13>
    (instrument) = UTENSIL)        <14>
```

Because of the mapping provided by "eats" between its subject and the ACTOR of EAT-FOOD, the restriction that this constituent's representation must be HUMAN is

propagated up to the NP which fills the SUBJ role specified by Eq. (5). Similarly, the FOOD restriction on the object of an EAT-FOOD would propagate to the NP assigned as the direct object (DOBJ) of "eats", because of Eqs. (11) and (13). The direct object is supplied by the following constraint rule:

```
(define-class VP
      (1) = V                                <15>
      (2) = NP                               <16>
      (head) = (1 head)                      <17>
      (head trans) = transitive              <18>
      (head dobj) = (2 head))                <19>
```

Using these rules, LINK constructs a DAG as its interpretation of both the structure and the meaning of a sentence during processing.

LINK is implemented in the style of a bottom-up chart parser.[2] However, whereas links in a standard chart are labeled with a syntactic category, in LINK's chart they are labeled with a DAG, representing all the syntactic and semantic features which have been assigned to each constituent.

New links are added to LINK's chart by applying unification rules such as the example rules above. Rules are indexed by subconstituents; i.e., by the values of paths which have numeric labels. Whenever a sequence of constituents in the chart is found which unifies with the corresponding sequence in a rule, that rule is applied. Unification guarantees that each constituent from the sentence has the syntactic and semantic features specified by the rule, or at least is compatible with those features.

Because of the use of a chart, it is possible for alternate interpretations of a constituent to be considered simultaneously. This occurs whenever the chart contains two or more links whose subconstituents overlap. Selection of an interpretation is in effect performed when a complete parse can be found which uses one of the competing links. As we will see, the ability to entertain multiple interpretations of a constituent is important in the interpretation of non-literal constructions.

4.2. LINK's Mapping Rules: An Overview

The definition of a mapping rule in LINK is shown below. A ⟨label⟩ is any syntactic or semantic category used in the grammar. A ⟨var⟩ (variable) by convention begins with a "?", and the same set of variables appear in the left- and right-hand sides of a rule.

⟨mapping-rule⟩ ::= ⟨spec⟩ ⇒ ⟨spec⟩

⟨spec⟩ ::= ⟨label⟩ ⟨equation⟩ . . . ⟨equation⟩

⟨equation⟩ ::= ⟨path⟩ = ⟨label⟩ | ⟨path⟩ = ⟨var⟩ |
⟨path⟩ = (⟨var⟩ ⟨label⟩) | ⟨path⟩ = ⟨path⟩

Mapping rules are interpreted to mean the following: whenever the parser builds a node with the appropriate label, and that node explicitly satisfies all the constraints specified in the left-hand-side constraint list, then an alternate interpretation can be built; namely, a node satisfying the description on the right-hand side. Variables indicate mappings between the values of properties of the original node and properties of the alternate node.

[2]See Winograd (1987) for a description of chart parsing.

To illustrate, let us consider three different examples of mapping rules. First, here is a mapping rule for the highly productive CHANGE-ALTITUDE is CHANGE-VALUE metaphor:

CHANGE-ALTITUDE ⇒ CHANGE-VALUE:
   (change) = ?X   (change) = ?X
                      (object) = NUMBER

The simplicity of this rule reflects the productivity of the metaphor. Rather than just providing an ambiguous definition for one word, such as "rose," it automatically creates an alternate interpretation whenever the semantic content of any word (or group of words) is constructed that fits the LHS. Thus, sentences as different as "the temperature fell" and "The population of NYC is slowly creeping upward" are handled by the same rule. Additional semantic constraints in other rules would ensure selection of the appropriate interpretation, depending on the semantic category of the OBJECT. Thus "the plane is rising" would be interpreted literally, while "the temperature is rising" would be interpreted metaphorically.

Mapping rules in LINK may be used to process other non-literal constructions, such as metonymies and figures of speech. Here is a rule for a common metonymy often used in conjunction with the CHANGE-ALTITUDE is CHANGE-VALUE metaphor:

PHYS-OBJ: ⇒ ?X
   (assoc-num) = (?X NUMBER)

Many physical objects are used to refer to closely associated numbers, e.g., the stock market for the Dow Jones Industrial Average, or the thermostat for its temperature setting. This rule means that "stock market" can be interpreted as either the building or the DJIA.

Finally, here is a rule for the idiom "to go through the roof," meaning to go up rapidly:

VP: ⇒ VP:
   (head rep) = PIERCE   (head rep) = CHANGE-ALTITUDE
   (head object rep) = ROOF   (head rep change) = +
   (head obj word) = roof   (head rep change rate) = FAST
   (head inf) = go

In this example, as with many idioms, *none* of the literal constructs of the LHS are relevant to the non-literal interpretation, so there is no need for variables. This rule also demonstrates that syntactic and lexical features of the construct may be used as easily as the semantic. Thus, the correct amount of flexibility is allowed for the idiom. In this case, ROOF must appear as the semantic object of the action PIERCE, thus restricting this interpretation to active constructions and disallowing examples such as "the roof was gone through by the stock market." The rule also specifies that the lexical items "go" and "roof" must be used, disallowing examples such as "the stock market pierced the roof."

Now consider the following example:

The stock market went through the roof.

In parsing this sentence, LINK creates literal and non-literal interpretations (links on the chart) for both phrases ("the stock market" and "went through the roof") before trying to combine them to form a complete sentence. The competing interpretations for each constituent coexist in the chart until one of them is used to build a complete parse of the sentence. In this case, given the semantic constraint that the stock market building is IMMOVABLE, and thus cannot be a (semantic) object of verbs requiring motion, then only the triply non-literal interpretation will unify to create a complete interpretation for

the sentence. If we make no such restriction, then three complete parses will be found, and the sentence is truly ambiguous. This example also demonstrates the flexibility of the first mapping rule, as alternate word definitions would need to be quite convoluted to arrive at the CHANGE-VALUE in this sentence.

Mapping rules may also be used in LINK to process constructions not usually thought of as non-literal. Some constructions, because of the productivity they demonstrate, require a rule which operates on intermediate (i.e., "literal") semantic interpretations. Consider the following examples:

> Mary gave John a lecture.
> John received a warm welcome from Mary.
> Mary got a big introduction from John before her talk.
> John gave Mary a ride to the airport.

These are all examples of a common expression in English, in which the doing of some action to a recipient is expressed as "giving" the recipient that act. Just as with many other non-literal constructions, this expression is quite flexible, independent of syntactic construction and relatively independent of the particular lexical item. The mapping rule for this expression is the following:

$$\text{TRANSFER-CONTROL:} \Rightarrow ?X:$$
$$(\text{object}) = (?X \text{ ACTION}) \qquad (\text{object}) = ?Y$$
$$(\text{recip}) = ?Y$$

Suppose TRANSPORT and SCOLD are defined semantically as ACTION's, and we have the following definitions in our lexicon:

```
(define-class N
        (1) = ride                              <20>
        (head type) = COMMON                    <21>
        (head rep) = TRANSPORT                  <22>
        (head rep instrument) = CAR)            <23>

(define-class N
        (1) = lecture                           <24>
        (head type) = COMMON                    <25>
        (head rep) = SCOLD)                     <26>
```

Here we can see the interaction of syntax and semantics. The constraint that the (head type) path must be COMMON means that these nouns require determiners in order to make NP's, while the semantic structure of the underlying verbs as ACTION's ensures that these nouns will trigger the mapping rule if ever found in a VP whose meaning is TRANSFER-CONTROL. Thus, "John gave Mary a ride" will produce exactly the same parse as "John transported Mary in a car," and "Mary gave John a lecture" will result in the same semantic representation as "Mary scolded John."

4.3. LINK's Mapping Rules: A Deeper Look

Because of the other capabilities of LINK, the mapping rules have a slightly more complicated structure than shown in the overview above. There are two main points to consider. First, the DAG that triggers a mapping rule may have been built up from an arbitrarily long phrase, and thus have a very complicated structure involving many arcs. Since the entire range of such arcs cannot be predicted, no mapping rule can anticipate

all arcs that might be useful to keep (to map over to the right-hand side). Second, there are rules that add semantic informatioin to a DAG based on the label of its root node (such as the EAT-FOOD rule in Sect. 4.1). The information in these rules is very closely tied to the literal meaning of the word, and thus is unlikely to be useful when mapped over.

To solve the first problem, the DAG that subsumed the LHS to trigger the mapping rule is unified with the result of the construction of the RHS. This unification is equivalent to having a default operation of keeping all the arcs of the original DAG when mapping occurs. This method may lead to problems, however, when the DAG contains arcs that have the same name as some that are created in the RHS of the mapping rule. Thus in the implementation of a mapping rule there is a set of arcs defined that are specifically removed from the DAG before unifying with the RHS. This set is called the THROW field. It is also useful for preventing certain classes of arc that the user knows may be present.

To solve the second problem, the arcs that are generated by adding semantic information based on the semantic nature of a node in the DAG are tagged as "literal" when they are added to the DAG. These "literal" arcs are also removed along with all those of the THROW field that are present. Sometimes a mapping may be flexible enough to allow a literal arc to map across (e.g., the CHANGE arc of CHANGE-ALTITUDE is still relevant to the CHANGE-VALUE after mapping). Such arcs are listed in a set called the KEEP field.

Thus a mapping rule is defined as in the previous section, but the THROW and KEEP fields are added to allow smooth integration with the rest of LINK, and to provide the user with another dimension of flexibility. Any mapping rule may be triggered at any point during the unification involved in creating a new link for the chart, whenever a node is created whose DAG satisfies the LHS of the rule. When this occurs, the LHS is matched to the DAG to bind the variables, then the "literal" arcs and those whose names appear in THROW are removed from (a copy of) the DAG. Next those arcs whose names appear in KEEP are added back to the DAG. The RHS is built, and unified with the modified version of the DAG. This new DAG is inserted into a duplicate of the link that was being created, at exactly the same location as the original DAG that triggered the mapping rule, and both links are inserted into the chart as alternate possibilities.

To illustrate the point, consider the idiom "to go through the roof" defined in the previous section. None of the arcs of the matched DAG are useful, so KEEP would be empty, and THROW would include (head). This will eliminate all syntactic and semantic information from the DAG that triggered the rule. For the CHANGE-ALTITUDE to CHANGE-VALUE metaphor, most arcs will probably be useful, but KEEP will include the most vital ones: (change) and (direction). Other arcs that might occur are (speed) and (acceleration), but any such arc would automatically map over by default.

5. LINK's MODEL AND THE PSYCHOLOGICAL DATA

Let us return now to the psychological findings on processing of non-literal constructions. Recall the three major results that we summarized earlier:

1. Reading times for non-literal constructions are no slower than reading times for literal constructions.
2. Non-literal meanings appear to be computed even when expressions are intended to be taken literally.
3. Literal meanings are not primed after comprehension of a sentence containing a non-literal construction.

Recall that these results contradict the Standard Pragmatic Model, which also uses mapping rules. However, our theory differs from the Standard Model in two important ways. First, in our theory the rules are applied automatically, regardless of whether or not semantic or pragmatic failures are encountered while constructing literal meaning. Second, in our theory a mapping rule is applied *before* the entire literal meaning has been processed. That is, the rule is applied whenever it matches the partial literal interpretation of the sentence built so far. The mapping rule then constructs an alternate (non-literal) interpretation of the input that has been processed. The result is that more than one possible interpretation of the utterance is constructed, even before the entire literal meaning of the input has been computed. Mapping rules never eliminate possible interpretations, they instead add other interpretations as alternatives. This differs from the Standard Model, in which literal meaning is transformed into (i.e., replaced by) a non-literal interpretation.

Because of these differences, our model can also account for findings 1 and 2 above. First, literal and non-literal expressions will take the same amount of time to compute, because non-literal interpretations are always computed, even in contexts in which the literal meaning is the one which is intended. Second, the fact that non-literal interpretations are *automatically* constructed means that they will be constructed even in literal contexts. This accounts for the findings of (Glucksberg *et al.* 1982) and (Gibbs 1983), in which evidence was found that non-literal interpretations were constructed in literal contexts.

We still have to explain the third finding, namely that literal meaning is not primed after comprehension of a sentence containing a non-literal expression. This would seem to suggest that literal meaning has not been computed, which would contradict our theory. How can we account for this discrepancy? To answer this question, consider the situation in which an ambiguous word is encountered in an utterance. Swinney (1979) and Tanenhaus *et al.* (1979) have shown that all senses of an ambiguous word (even if it is syntactically ambiguous) are activated briefly (between 200 and 600 msecs), regardless of context, when the word is encountered, after which contextual information suppresses inappropriate senses. We envision that this same selection process is used in comprehension of non-literal interpretations. After a mapping rule has been applied, the situation is analogous to one in which all senses of an ambiguous word have been activated. As possible alternative interpretations are computed, contextual information constrains which interpretations are acceptable in that context. Thus, by the end of a sentence containing a nonlilteral expression, it is quite likely that the literal sense of the expression has already been suppressed by contextual constraints, just as alternate meanings of an ambiguous word are suppressed by context. This suggests that Gibbs's (1983) results indicating that literal meanings were not primed at the end of a sentence may have been due to the target appearing after suppression of the literal meaning had already taken place.

This explanation does not yet account for the assymmetry of Gibbs's results. Recall that Gibbs found that non-literal meanings were still primed at the end of a sentence which was interpreted literally. It is possible that the asymmetry of these results could be explained by frequency effects: if the indirect request sense of a construction occurs much more frequently than the literal sense, this might overpower the ability of context to suppress this meaning.

Cacciari and Tabossi's (1988) results also require further explanation. Recall that for familiar idioms, literal meaning of the final word was not activated, even immediately after hearing the final word. This suggests that if an idiom is familiar enough, subjects have already selected the idiomatic meaning as the preferred interpretation even before hearing the entire idiom. Thus, the idiom is triggered by a subset of the words in the idiom, rather than the entire construction. This is consistent with the general framework of our mapping rule theory. It simply suggests that, in cases where the non-literal construction is familiar

enough, the constraints of the appropriate mapping rule have been loosened so that it is triggered with incomplete information; i.e., before the entire literal semantic content of the idiomatic expression has been constructed. If this is the case, a mapping from literal to non-literal meaning is still an essential part of the comprehension process, but only a portion of the literal meaning is used to trigger the mapping. Thus, we would predict that the literal meaning of earlier words in an idiom should be found to be activated if a similar priming experiment were performed with appropriate targets placed in the middle of the idiomatic expressions.

6. CONCLUSION

The psychological evidence that we have cited above clearly indicates that figurative meanings are processed as quickly as literal meanings, given an appropriate context. This evidence has shown the Standard Pragmatic Model to be an inadequate model of human processing of non-literal constructions. Since the Standard Pragmatic Model stipulates that the first step in understanding an utterance is to compute its literal meaning, and since the Standard Pragmatic Model's failure to explain the psychological data is directly tied to this first step in understanding, the role of literal meaning in understanding has come into question. Many researchers, Gibbs among them, have claimed that the evidence shows that literal meaning need not be computed on the way to computing intended meaning.

We have argued that the problem with the Standard Pragmatic Model lies not in the fact that it computes literal meaning, but in the process by which literal meaning is computed and used to compute figurative meaning. We have presented an alternative model of semantic interpretation which computes literal meaning even for non-literal constructions, but which is consistent with the existing psychological data.

Although we feel that there are strong computational motivations for preferring mapping rules (see Sect. 2), it appears that the existing empirical evidence is not adequate to distinguish between Gibbs's claim that literal meaning is not computed and our alternate model. The data are inadequate because, in general, they only tell us about processing at the sentence level, rather than the word or phrase level. The reading time comparisons which we have discussed have been done on entire sentences rather than on individual words or phrases, and priming experiments have tested facilitation only at the end of sentences. Thus, existing results say little about the process of comprehension of non-literal expressions at the word or phrase level. It is at this level that we could distinguish between the competing theories.

What evidence would shed additional light on the matter? Recall that in our model, when a potentially non-literal expression is processed, the lexicon is accessed (to get the literal meaning of words/phrases), and then mapping rules are used to compute the non-literal interpretation of the utterance. Our model, then, predicts that the activation levels of literal and non-literal meaning should follow the pattern shown in Fig. 2. Upon encountering a non-literal expression, the lexicon is accessed, resulting in the activation of the literal meaning at time t1. A short time later at time t2, after application of the mapping rule is complete, the non-literal meaning is also activated. Finally, at time t3, context determines the appropriate meaning and suppresses the inappropriate one. These predictions are quite similar to the findings of Cacciari and Tabossi on the comprehension of unfamiliar idioms (see Sect. 3.2). On the other hand, Gibbs's model would presumably predict that the literal sense of a non-literal expression should not be activated, even temporarily, in a non-literal context, since he asserts that literal meaning is not computed in these contexts.

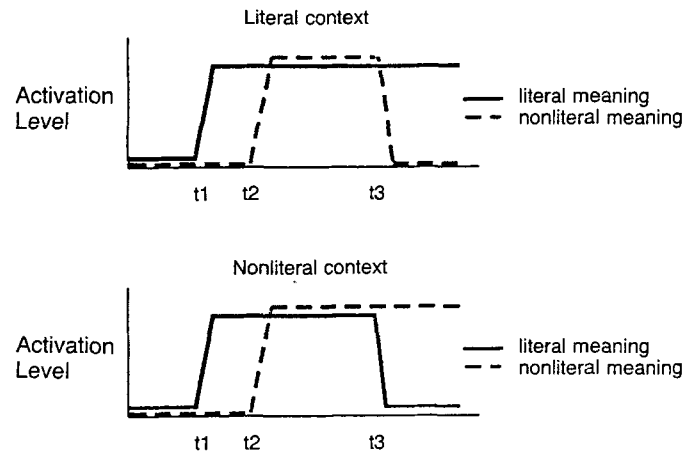Literal context

Nonliteral context

FIGURE 2. Predicted activations of literal and non-literal meanings over time in different contexts.

The predictions of our model are analogous to the work of Swinney (1979) and Tanenhaus *et al.* (1979), as well as many other experiments on lexical disambiguation (see Simpson 1984 or Small *et al.* 1988 for a review of this research). Our theory predicts similar effects for non-literal constructions: regardless of whether a particular construction is meant literally or not, we would expect both interpretations to be active for a brief period.

ACKNOWLEDGMENTS

REFERENCES

BALLIM, A., Y. WILKS, and J. BARNDEN. 1991. Belief ascription, metaphor, and intensional identification. Cognitive Science, 15: 133–171.

CACCIARI, C., AND P. TABOSSI, (1988). The comprehension of idioms. Journal of Memory and Language, 27: 668–683.

CARBONELL, J. 1982. Metaphor: An inescapable phenomenon in natural language comprehension. *In* Strategies for Natural Language Processing. *Edited by* W. Lehnert and M. Ringle. Lawrence Erlbaum Associates, Hillsdale, NJ. pp. 415–434.

FASS, D. 1988. An account of coherence, semantic relations, metonymy, and lexical ambiguity resolution. *In* Lexical ambiguity resolution: Perspectives from psycholinguistics, neuropsychology, and artificial intelligence. *Edited by* S. Small, G. Cottrell, and M. Tanenhaus. Morgan Kaufmann, San Mateo, CA.

GERRIG, R. 1989. Constraints on theories of metaphor. Cognitive Science, 13: 235–241.

GIBBS, R. 1979. Contextual effects in understanding indirect requests. Discourse Processes, 2: 1–10.

GIBBS, R. 1983. Do people always process the literal meanings of indirect requests? Journal of Experimental Psychology: Learning, Memory, and Cognition, 9: 524–533.

GIBBS, R. 1984. Literal meaning and psychological theory. Cognitive Science, 8, pp. 275–304.

GILDEA, P., AND S. GLUCKSBERG. 1983. On understanding metaphors: The role of context. Journal of Verbal Learning and Verbal Behavior, 22 pp. 577–590.

GLUCKSBERG, S., P. GILDEA, and H. BOOKIN. 1983. On understanding non-literal speech: Can people ignore metaphors? Journal of Verbal Learning and Verbal Behavior, 22, pp. 577–590.

INDURYKA, B. 1987. Approximate semantic transference: A computational theory of metaphors and analogies. Cognitive Science, 11, pp. 445–480.

INHOFF, A., S. LIMA, and P. CARROLL. 1984. Contextual effects on metaphor comprehension in reading. Memory and Cognition, 12(6): 558–567.

KATZ, J., and J. FODOR. 1963. The structure of a semantic theory. Language, 39: 170–210.

LAKOFF, G., and M. JOHNSON. 1980. Metaphore we live by. University of Chicago Press, Chicago, IL.

LANGACKER, R. 1987. Foundations of cognitive grammar. Vol. 1. Stanford University Press, Stanford, CA.

MARTIN, J. 1990. A computational model of metaphor interpretation. Academic Press, San Diego, CA.

ORTONY, A., D. SCHALLERT, R. REYNOLDS, and S. ANTOS. 1978. Interpreting metaphors and idioms: Some effects of context on comprehension. Journal of Verbal Learning and Verbal Behavior, 17: 465–477.

POLLARD, C., and I. SAG. 1987. Information-based syntax and semantics. Center for the Study of Language and Information, Menlo Park, CA.

SEARLE, J. 1978. Literal meaning. Erkenntnis, 13: 207–224.

SHIEBER, S. 1986. An introduction to unification-based approaches to grammar. CSLI, Stanford, CA.

SIMPSON, G. 1984. Lexical ambiguity and its role in models of word recognition. Psychological Bulletin, 96(2): 316–340.

SMALL, S., G. COTTRELL, and M. TANENHAUS. 1988. Lexical ambiguity resolution: Perspectives from psycholinguistics, neuropsychology, and artificial intelligence. Morgan Kaufmann, San Mateo, CA.

SWINNEY, D. 1979. Lexical access during sentence comprehension: (Re)consideration of context effects. Journal of Verbal Learning and Verbal Behavior, 18: 645–659.

TANENHAUS, M., J. LEIMAN, and M. SEIDENBERG. 1979. Evidence for multiple stages in the processing of ambiguous words in syntactic contexts. Journal of Verbal Learning and Vebal Behavior, 18: 427–440.

WINOGRAD, T. 1987. Language as a cognitive process. Vol. 1: Syntax. Addison-Wesley, Reading, MA.