

Conventional Metaphor and the Lexicon

James H. Martin

*Computer Science Department and
Institute of Cognitive Science
University of Colorado,
Boulder, CO
80309-0430
martin@cs.colorado.edu*

Abstract

Metaphor and other forms of non-literal language are essential parts of language which have direct bearing on theories of lexical semantics. Neither narrow theories of lexical semantics, nor theories relying solely on world knowledge are sufficient to account for our ability to generate and interpret non-literal language. This paper presents an emerging approach that may provide such an account. This approach is based on systematic representations that capture non-literal language conventions, and mechanisms that can dynamically understand and learn new uses as they are encountered.

1 Introduction

In traditional approaches to Lexical Semantics a whole host of phenomena have been largely banished from consideration. Among the unwelcome phenomena have been irony, sarcasm, metaphor, metonymy, indirect requests, and productive idioms. These phenomena have been relegated to various areas of pragmatics, context, and world knowledge. Implicit in this banishment is the assertion that the mechanisms and knowledge necessary to account for these uses are fundamentally different from those used in ordinary semantic interpretation. Moreover, this additional knowledge is generally held to be that of a more general common-sense conceptual kind.

This point of view has a number of useful properties. Firstly, it severely limits the range of uses that need to be accounted for by lexical semantics. Therefore, the sheer amount of work faced by the lexical semanticist can be reduced. The job of lexical semantics is essentially finished once the semantics of a word's literal meaning have been adequately represented and tied to the observed syntactic behavior of the word.

The second important property (really just a different view of the first) is the notion that a lexicon with a relatively small fixed number of senses per word can still hope to account for the much larger number of senses actually encountered in practice. This follows from the fact that the range of senses is tied to the amount of real world knowledge in the knowledge-base, and the ability of the extra-linguistic mechanisms to make use of this knowledge. It is this second property that is most relevant to approaches to computational lexical semantics. This state of affairs can be summarized as follows.

- Lexical semantics is limited to individual lexical items paired with literal meanings.
- Non-literal uses are dealt with by mechanisms that are not required for otherwise "normal" semantic processing. (Among the methods that have been used are analogical matchers, planners, and spreading activation.)

- Performance (the number of senses that can be accounted for) is dependent on the amount of general world knowledge.

Unfortunately, as attractive as this formulation may be, a number of serious problems call it into question. These problems indicate that neither a narrowly construed version of lexical semantics, nor large amounts of world knowledge, nor a combination of the two, can account for the linguistic competence required to deal with non-literal language. The following two sections will outline two problems with the traditional formulation and its computational counterparts.

2 Constraints from Psycholinguistic Research

While it is difficult to apply results from psycholinguistics to semantic theories in a direct fashion, these results can nevertheless pose useful rough constraints. The results that are of interest here stem from research on the relative difficulty of understanding what has been called literal language versus various kinds of metaphorical, idiomatic, and indirect language [Gerrig, 1989, Gibbs, 1984, Gibbs, 1989, Ortony *et al.*, 1978].

The basic result that will be used is that the time needed to process various kinds of non-literal language does not differ significantly from the time taken to interpret direct language in the appropriate context. Specifically, there is no experimental evidence to indicate that there is a radical difference between the time taken to interpret metaphorical language and that taken to interpret direct literal language. This constraint has been referred to as the *total time constraint* [Gerrig, 1989].

This rough equivalence of time to process must be taken into account by any realistic approach to semantic interpretation. While the empirical result of equivalent time to process does not necessarily imply that similar mechanisms are at work, in the absence of more fine-grained empirical results indicating that fundamentally different processes are at work, it seems reasonable to assume that the mechanisms will be similar. At the very least it must be shown that if different processes are at work then they should have roughly comparable computational complexity. The next section will show that the proposed methods for metaphor do not meet this criteria.

3 Problem-Solving Approaches

The major approaches to dealing with the interpretation of non-literal language can all be characterized as “problem-solving” approaches. These include plan-based and analogy-based methods that attempt to find or infer the most likely meaning for a non-literal use when it is encountered [Carbonell, 1981, DeJong and Waltz, 1983, Fass, 1988, Gentner *et al.*, 1988, Indurkha, 1987]. In the case of metaphor, these approaches assert that metaphors arise from an underlying conceptual similarity or analogy between the concepts representing the literal meaning of the words and the concepts underlying the ultimate meaning of the utterance. They make no use of knowledge about the conventions of the language. The task of interpreting metaphoric language is seen as a special purpose problem-solving task requiring access to knowledge and inference techniques that are not otherwise a part of the normal language processing faculties.

Note that the computational costs of these analogy mechanisms are radically higher than those posed for direct non-metaphorical language. While the details of each approach

differ, they are all fundamentally based on a two stage model where the literal meaning of the sentence is computed and judged to be ill-formed, and then an analogy system is employed to search for an appropriate target meaning. The major problem with these approaches is intractability. The proposed methods are for most practical purposes computationally intractable given any realistic representation of the domain. The problem is particularly acute when the system is intended to be used in some kind of user interface role where response time is critical. Moreover, such methods are directly contradicted by the empirical psycholinguistic evidence described above.

4 An Emerging View

These psycholinguistic and computational results have led to a surge of computational research in the area of semantic interpretation of non-literal language in recent years. This research has resulted in a number of systems that can handle one or more of the following phenomena: metaphor, metonymy, indirect requests, and somewhat productive phrasal idioms. [Fass, 1988, Hinkelman, 1989, Martin, 1990, Pustejovsky, 1991, Zernik, 1987] A careful analysis of these disparate efforts reveals that a common approach to the semantic interpretation of non-literal language may be emerging. At the heart of this approach is a concern with the tension between the conventionality of many non-literal uses, and the ability to understand and generate new uses that go beyond static conventions. This concern has generally led to a two-part approach involving systematic representations that capture conventions, and mechanisms that can dynamically understand and learn new uses as they are encountered.

The work of Pustejovsky [Pustejovsky, 1991] is related in its attempt to reduce the size and complexity of individual lexical entries. The difference is that while the above efforts try to achieve a kind of lexical generativity through the use of extra-lexical and extra-grammatical language conventions, Pustejovsky's work can be seen as attempting to capture this generativity from within a lexical knowledge base.

Consider some of the following examples.

- (1) How can I *kill* a process?
- (2) My emacs just *died*.
- (3) Can you open the door?
- (4) The buses are going on strike.

Each of these examples makes use of a completely conventional, yet non-literal, way of conveying the intended meaning. The use of the word *kill* in (1) is consistent with the conventional and productive metaphor that allows computer processes to be viewed as living-things. This productive use also motivates the use of *die* in (2). Example (3) makes use of a conventional indirect way to make a request in English. In this case, asking about one's ability to perform some task by using *Can you*, indicates the speaker's desire to have that task performed by the hearer. Finally, the last example demonstrates the widespread use of metonymy in English. In this example, the word *bus* is used to refer to the distinct, yet obviously related, busdrivers.

The unifying theme, in all of these examples, is the use of conventional yet productive mechanisms to produce these sentences. The remainder of this paper will introduce a unifying approach to these and similar phenomena that avoids some of the problems inherent in current approaches.

The following sections provide an analysis that sketches out the nature of this emerging approach. In each of these sections I will first attempt to abstractly characterize the approach and then give a few details of how the approach has been instantiated in the MIDAS [Martin, 1990] system. To be specific, Section 5 describes the nature of the knowledge that these systems use, and introduces a new knowledge construct called a *Conventional Conceptual Mapping*, that abstractly accounts for them all. Section 7 briefly describes how this approach has been applied to metaphor in MIDAS.

5 Conventional Conceptual Mappings

The fundamental construct that all these systems use is a knowledge structure that allows the system, under certain conditions, to map from one set of concepts to another set of concepts. In the case of MIDAS, these associations are used to represent the source and target concepts of conventional metaphors; in [Fass, 1988] they are used to capture the two component parts of metonymies; while in [Hinkelman, 1989] they are used to capture knowledge about indirect speech acts. Consider the following examples.

- (5) How can I get out of Emacs?
- (6) Can you speak Spanish?
- (7) Plato is on the top shelf.

In Example (5), MIDAS uses a set of mappings from the domain of enclosures to the domain of computer processes to process this question. In (6), Hinkelman's system may be viewed as making use of structured associations that link features and concepts associated with this surface form, to a concept representing a request to perform the intended meaning. In this example, this association links questions pertaining to the ability to perform an action to requests for that action to be performed. Finally, in (7), Meta5 makes use of mappings that represent conventional metonymies. In this case, the system makes use of a mapping that allows certain kinds of products to be referred to through their producer. In this case, the works of an author can be referred to by referring to the author.

There are a number of common characteristics that are shared among these efforts. The most important of which is the idea that these associations may be represented at a variety of levels ranging from linguistic features to concepts. This has a number of important ramifications. The first is that it allows these mappings to capture the generalizations at the right level of conceptualization. For example, it is not necessary in (7) to have a new word sense for the word Plato. Rather, the system merely needs to know the fact that Plato was an author to use the proposed metonymic map linking producers to products. It is also important to note that while these conventional associations interact with world knowledge, they can not be conflated with it.

The second important characteristic shared among these efforts is the idea that these structured associations have explicitly represented constraints on their use. When the constraints on the use of the association can be satisfied in context, then the association can be used to map from one set of concepts to another.

I call a mapping that has these characteristics a *Conventional Conceptual Mapping* (CCM). More specifically, a CCM is a structured mapping that has the following characteristics.

- It allows disparate sets of concepts to be associated in a structured manner.

- It can be represented at varying degrees of abstraction.
- It can have constraints on its use explicitly represented.

Not surprisingly, none of the systems mentioned above fits this description precisely. Most deal with only one of the phenomena, while ignoring the others. For example, Hinkelman's system deals only with speech act level conventions, while Martin's MIDAS system deals only with conventional metaphors. While Fass's META5 system does handle metaphor and metonymy, only metonymy is handled in a way that uses CCM's.

The following section will give a high level discussion of how the this approach can be applied to the problem of metaphor. Section 7 will then detail the current state of an implementation of this approach.

6 A Sample Metaphor Analysis

To make this discussion more concrete, consider some of these issues in terms of the following examples.

- (8) I *came* to the conclusion that Mike was right.
- (9) I *reached* the conclusion that Mike was right.
- (10) I *arrived* at the conclusion that Mike was right.
- (11) I was *led* to the conclusion that Mike was right.
- (12) I was *dragged kicking and screaming* to the conclusion that Mike was right.
- (13) That Mike was right *came* to me.
- (14) It *came* to me that Mike was right.
- (15) It *hit* me that Mike was right.
- (16) It *struck* me that Mike was right.
- (17) Mike *struck* me as being right.
- (18) Mike *came across* as being right.

Each of these examples expresses the idea that someone is changing state with respect to some belief. More specifically, they express the idea that the believer now believes some proposition that had not been previously held. They all express this notion through the use of a spatial metaphor that has the following entailments: beliefs are external objects with locations, believers have locations, shared location between a believer and a proposition indicates belief, and finally movement to a shared location indicates a state-change resulting in belief. [Barnden, 1989]

Given this general metaphor, we can narrow the discussion further by isolating on the behavior of a single lexical item.

- (19) The idea that Mike was smart *came* to me.
- (20) It *came* to me that Mike was smart.
- (21) I *came* to the conclusion that Mike was smart.
- (22) ?I *came* to the idea that Mike was smart.
- (23) I *came* upon the idea that Mike was smart.

These examples illustrate at least two broad semantic senses and four different syntactic environments for the word *come*. This leads to the big question:

Do these meanings, and these syntactic environments, have to be listed in the lexical entry for *come*, or can these facts be motivated by other independent language conventions?

The following discussion will raise a series of more specific issues surrounding this question and provide some flavor of the research program we are pursuing.

Consider first the two broad senses that are evident here. In the first sense, the new proposition is the mover, changing its location to the location of the believer. One implication of this sense is that the believer did not consciously expend any effort to produce this belief. This sense is illustrated in (19) and (20). Examples (21) through (23) illustrate the second sense that has the believer actively moving to the new proposition. This sense has more of an implication that the believer is taking an active part in the process.

We are currently investigating the proposition that these two senses result from the interaction of a single broad spatial metaphor, the core spatial semantics of the word *come*, and facts about the situation. It is, therefore, not necessary to have either two lexical senses or two conventional metaphors.

Now consider the syntactic variation exhibited in examples (19) and (20). In both of these examples the proposition is in motion. In (19) it appears in the subject position, while in (20) it is extraposed. Our current hypothesis is that it is not necessary to list in the lexical entry for *come* the fact that it permits extraposition. Rather this flows from the general requirements of the extraposition construction combined with the semantics of target concept of this metaphor.

Next consider the apparent well-formedness variation between (21) and (22). The use of *come to* with *conclusion* is clearly preferred. Again the metaphorically focussed approach advocated here would say that the metaphorical semantics of the word *conclusion* lead it to naturally fit into the spatial metaphor governing the use of *come*.

Finally, consider the difference between example (23) and (21). This example seems to imply that, while the believer was in motion towards the proposition, the discovery was in fact something of a chance occurrence. But of course, this would also be implied by a literal use of *come upon*. Therefore, this difference is attributable to a difference in the spatial semantics of *come*. The question remains as to whether it is necessary to list this fact in the lexical entry for *come*.

7 Overview of MIDAS

This section provides a brief overview of our current computational realization of this approach to metaphor. In particular, it introduces the following issues.

Representation: The explicit representation of the conventional metaphors in a language in the form of explicit associations between concepts.

Interpretation: The correct and efficient application of this metaphoric knowledge to the interpretation of metaphoric language.

Under this view, the proper way to approach the study of metaphor is to study the underlying details of individual metaphors and systems of metaphors in the language. This approach follows on the metaphor work of Lakoff and Johnson [Lakoff and Johnson, 1980] and the computational approaches to metaphor described in [Jacobs, 1985, Norvig, 1987].

This approach has been embodied in MIDAS (Metaphor Interpretation, Denotation, and Acquisition System) [Martin, 1990] MIDAS is a set of computer programs that can be used to perform the following tasks: explicitly represent knowledge about conventional metaphors, apply this knowledge to interpret metaphoric language, and learn new metaphors as they are encountered.

7.1 Knowledge Representation

Consider the following simple example of a conventional UNIX metaphor. The metaphorical use of the word *in* reflects a systematic metaphorical structuring of computer processes as enclosures.

(24) I am *in* Emacs.

Metaphors like this may be said to consist of the following component concepts: a *source* component, a *target* component, and a set of conventional associations from the source to target. The target consists of the concepts to which the words are actually referring. The source refers to the concepts in terms of which the intended target concepts are being viewed. In this example, the target concepts are those representing the state of currently using a computer process. The source concepts are those that involve the state of being contained within some enclosure.

The approach taken here is to explicitly represent conventional metaphors as sets of associations between source and target concepts. The metaphor specifies how the source concepts reflected in the surface language correspond to various target concepts. In this case, the metaphor consists of component associations that specify that the state of being enclosed represents the idea of currently using the editor, where the user plays the role of the enclosed thing, and the Emacs process plays the role of the enclosure. These associations also serve to delimit the particular parts of the various source and target domains that are relevant to particular conventional metaphors.

Note also that these source-target associations are represented at the conceptual and not the lexical level. Any single lexical item or expression that can be construed as referring to the source concept of a known metaphor, may invoke that metaphor. In this example, the source component of the metaphor is attached to the concept of being enclosed, not to the lexical item *in*.

These sets of metaphoric associations, along with the concepts that comprise the source and target domains, are represented using the KODIAK [Wilensky, 1986] representation language. KODIAK is an extended semantic network language in the tradition of KL-ONE [Brachman and Schmolze, 1985] and its variants.

These sets of metaphoric associations representing conventional metaphors are full-fledged KODIAK concepts. As such, they can be related to other concepts and arranged in abstraction hierarchies using the inheritance mechanisms provided by KODIAK. The hierarchical organization of conventional metaphoric knowledge is the primary means used to capture the regularities exhibited by the system of metaphors in the language. Specifically, KODIAK is used to represent specialized domain specific metaphors, pervasive high-level metaphors, and the systems of relations among related metaphors.

7.2 Interpretation

The interpretation process in MIDAS is basically one that views a given input sentence as providing a set of constraints on possible interpretations. MIDAS checks the input constraints against all the possible interpretations that can be conventionally associated with the input. Interpretations that are coherent with the constraints are returned. The possible conventional interpretations may include direct non-metaphoric interpretations, as well as all the conventional metaphors that are invoked by the input.

Consider the details of the following shortened trace. In this example, MIDAS must find a coherent interpretation for this use of *enter*. MIDAS finds, and attempts to apply, all the conventional metaphorical and non-metaphorical concepts associated directly with, or inherited by, this concept. In this case, it finds that the only conventional interpretation that is consistent with the input is the one that results from the application of the known *Enter-Lisp* metaphor.

```
> (do-sentence)
Interpreting sentence:

How can I enter lisp?

Interpreting concreted input.

(A Entering50 (↑ Entering)
 (enterer50 (↑ enterer) (A I203 (↑ I)))
 (entered50 (↑ entered) (A Lisp58 (↑ Lisp))))
```

A parser first produces a syntactic analysis and a preliminary semantic representation of the input. At this point in the analysis, UC calls upon MIDAS to begin a deeper analysis of this initial representation.

```
Failed interpretation: Entering50 as Entering.
Failed interpretation: Entering50 as Enter-Association.
Valid known metaphorical interpretation: Entering50 as Enter-Lisp.
```

The case structure of this preliminary representation is checked against the semantic constraints of all the interpretations conventionally associated with the *Entering* concept. In this case, MIDAS finds that the direct interpretation and one of the other possible entering metaphors can be rejected before the appropriate *Enter-Lisp* metaphor is found.

It is important to realize that the order of the search performed here is arbitrary. MIDAS is exhaustively finding all conventional interpretations that are consistent with the input. The determination of consistency for any given interpretation is independent of the consistency of any of the other possible interpretations. In particular, the well-formedness of a direct, or literal, interpretation has no effect on whether or not a metaphorical interpretation will be found. It follows from this that the order of the search through the possible interpretations has no effect on which interpretations will ultimately be produced.

Applying conventional metaphor Enter-Lisp.

```
(A Enter-Lisp (↑ Container-Metaphor Metaphor-Schema)
  (enter-lisp-res enter-res → lisp-invoke-result)
  (lisp-enterer enterer → lisp-invoker)
  (entered-lisp entered → lisp-invoked)
  (enter-lisp-map Entering → Invoke-Lisp))
```

Mapping input concept Entering50
to concept Invoke-Lisp30

Mapping input role enterer50 with filler I203
to target role lisp-invoker30

Mapping input role entered50 with filler Lisp58
to target role lisp-invoked30

Yielding interpretation:

```
(A Invoke-Lisp30 (↑ Invoke-Lisp)
  (lisp-invoked30 (↑ lisp-invoked) (A Lisp58 (↑ Lisp)))
  (lisp-invoker30 (↑ lisp-invoker) (A I203 (↑ I))))
```

MIDAS then begins the process of mapping from the given source concepts to the appropriate target concepts based on the constraints imposed by the metaphor. The mapping process, called *metaphoric unviewing*, creates a new instance of the metaphor itself along with the attendant source and target concepts. In this example, the source concept of **Entering** is mapped to the target concept **Invoke-Lisp** as specified by the metaphor.

Final Representation:

```
(A How-Q207 (↑ How-Q)
  (topic206 (↑ topic)
    (A Invoke-Lisp30 (↑ Invoke-Lisp)
      (lisp-invoked30 (↑ lisp-invoked) (A Lisp58 (↑ Lisp)))
      (lisp-invoker30 (↑ lisp-invoker) (A I203 (↑ I))))))
```

8 Conclusions

The competence required for the frequent and conventional use of idiom, metaphor, metonymy, irony, and sarcasm can not be accounted for by either a narrow theory of lexical semantics or any theory relying on world knowledge alone. Moreover, empirical evidence suggests that these phenomena constitute an integral part of our linguistic competence. It is therefore clear that our semantic theories must be broadened beyond their current scope to adequately deal with the challenges posed by these phenomena. The work outlined in this paper is a start in this direction.

References

- [Barnden, 1989] John Barnden. Belief, metaphorically speaking. In *Proceedings of the First International Conference on Principles of Knowledge Representation*, San Mateo, CA, 1989. Morgan Kaufmann.
- [Brachman and Schmolze, 1985] Ronald J. Brachman and James Schmolze. An overview of the kl-one knowledge representation system. *Cognitive Science*, 9:346–370, 1985.
- [Carbonell, 1981] Jaime Carbonell. Invariance hierarchies in metaphor interpretation. In *Proceedings of the Third Meeting of the Cognitive Science Society.*, pages 292–295. Cognitive Science Society, August 1981.
- [DeJong and Waltz, 1983] Gerald F. DeJong and David L. Waltz. Understanding novel language. *Computers and Mathematics with Applications*, 9, 1983.
- [Fass, 1988] Dan Fass. *Collative Semantics: A Semantics for Natural Language*. PhD thesis, New Mexico State University, Las Cruces, New Mexico, 1988. CRL Report No. MCCS-88-118.
- [Gentner *et al.*, 1988] D. Gentner, B. Falkenhainer, and J. Skorstad. Viewing metaphor as analogy. In D.H. Helman, editor, *Analogical Reasoning*. Kluwer Academic Publishers, 1988.
- [Gerrig, 1989] Richard J. Gerrig. Empirical constraints on computational theories of metaphor: Comments on Indurkha. *Cognitive Science*, 13(2):235–241, 1989.
- [Gibbs, 1984] Raymond W. Gibbs. Literal meaning and psychological theory. *Cognitive Science*, 8:275–304, 1984.
- [Gibbs, 1989] Raymond W. Gibbs. Understanding and literal meaning. *Cognitive Science*, 13(2):243–251, 1989.
- [Hinkelman, 1989] Elizabeth Hinkelman. *Linguistic and Pragmatic Constraints on Utterance Interpretation*. PhD thesis, University of Rochester, Rochester, NY, 1989. Technical Report No. UR-CS-288.
- [Indurkha, 1987] Bipin Indurkha. Approximate semantic transference: A computational theory of metaphors and analogy. *Cognitive Science*, 11:445–480, 1987.
- [Jacobs, 1985] Paul S. Jacobs. *A Knowledge-Based Approach to Language Production*. PhD thesis, University of California, Berkeley, Computer Science Department, Berkeley, CA, 1985. Report No. UCB/CSD 86/254.
- [Lakoff and Johnson, 1980] George Lakoff and Mark Johnson. *Metaphors We Live By*. University of Chicago Press, Chicago, Illinois, 1980.
- [Martin, 1990] James H. Martin. *A Computational Model of Metaphor Interpretation*. Academic Press, Cambridge, MA, 1990.
- [Norvig, 1987] Peter Norvig. *A Unified Theory of Inference for Text Understanding*. PhD thesis, University of California, Berkeley, Computer Science Department, Berkeley, CA, 1987. Report No. UCB/CSD 87-339.

- [Ortony *et al.*, 1978] A. Ortony, D. Schallert, R. Reynolds, and S. Antos. Interpreting metaphors and idioms: Some effects of context on comprehension. *Journal of Verbal Learning and Verbal Behavior*, 17:465-477, 1978.
- [Pustejovsky, 1991] James Pustejovsky. Towards a generative lexicon. *Computational Linguistics*, 17(1), 1991.
- [Wilensky, 1986] Robert Wilensky. Some problems and proposals for knowledge representation. Technical Report UCB/CSD 86/294, University of California, Berkeley, Computer Science Division, May 1986.
- [Zernik, 1987] Uri Zernik. *Strategies in Language Acquisition: Learning Phrases from Examples in Context*. PhD thesis, University of California, Los Angeles, Computer Science Department, Los Angeles, CA, 1987.