Metaphor comprehension: A computational theory

Walter Kintsch

Department of Psychology
University of Colorado
Boulder, CO 80309-0344
(303)-492-8663
wkintsch@psych.colorado.edu

In press, *Psychonomic Bulletin & Review*

Abstract

Metaphor comprehension involves an interaction between the meaning of the topic and vehicle terms of the metaphor. Meaning is represented by vectors in a high-dimensional semantic space. Predication modifies the topic vector by merging it with selected features of the vehicle vector. The resulting metaphor vector can be evaluated by comparing it with known landmarks in the semantic space. Thus, metaphorical predication is treated in the present model in exactly the same way as literal predication. Some experimental results concerning metaphor comprehension are simulated within this framework, such as the non-reversibility of metaphors, priming of metaphors with literal statements, and priming of literal statements with metaphors.

The rich body of experimental results that has appeared in the psychological literature in recent years (for reviews, see Cacciari & Glucksberg, 1994; Gibbs, 1994a) has changed our understanding of how non-literal statements such as metaphors are comprehended. Prior to that work the dominant view was that the comprehension of non-literal statements involves two steps: first, it must be recognized that the statement makes no sense if interpreted literally; then its intended, non-literal meaning is computed by some kind of inference. Now we know that, instead, metaphors can be understood directly, like literal statements. A computational model of literal comprehension should therefore be able to understand metaphorical statements in the same way that it "understands" literal sentences.

In this paper I shall sketch a computational model of metaphor comprehension that treats metaphors in the same way as literal statements. I introduce this model with an example that Glucksberg used to present his view that metaphorical predication is basically the same as literal predication (Glucksberg, 1998). According to Glucksberg, the metaphor *My lawyer is a shark* is a regular class-inclusion assertion, except that "the metaphor vehicle (*shark*) is used to refer to the superordinate category of predatory creatures in general, not to the smaller, concrete category of marine creatures that is also named *shark*" (Glucksberg, 1998, p. 41). Thus, the metaphorical *shark*-properties - *vicious, predatory, aggressive*, and *tenacious* - are attributed to *lawyer*, but the literal *shark*-properties - *fast swimmer, has fins, has sharp teeth, has leathery skin, has gills* – are not. The goal of the present paper is to show how such a process can be realized computationally.

Glucksberg's discussion fairly summarizes the empirical evidence on metaphor comprehension, but is incomplete in one important way: how do we know what is a superordinate-category level and what is a basic-level property? After all, the basic-level *shark* is a member of several superordinate categories, and Glucksberg's intuitive choice of the right one (*predatory creatures*, instead of, for example, *fish*) is unsatisfactory from a computational standpoint. A model of comprehension must select the right features automatically, without having to be told what is relevant and what is not.

Metaphorical Predication[1]

If metaphors are understood by people in much the same way as literal sentences, then metaphorical predication becomes a special case of predication in general. In this paper, a general computational theory of predication, which has recently been proposed by Kintsch (submitted), will be applied to simulate metaphor comprehension. This theory has two basic components: a model of human knowledge structure provided by Latent Semantic Analysis (LSA), and a model of text comprehension, the construction-integration (CI) model. LSA is a method for automatically constructing a high-dimensional semantic space from the analysis of a large amount of written text. An introduction and further references are given by Landauer, Foltz, and Laham (1998). The CI model is a psychological model of text comprehension that has been applied in a wide variety of situations (Kintsch, 1988; 1998). The theory presented below is an extension and elaboration of Kintsch (1998, Chapter 5) and introduces a new way of modeling predication within the context of the CI – LSA framework (Kintsch, submitted).

LSA (Landauer & Dumais, 1997) is a contextual theory of meaning in that it represents the meaning of a word by its relationships to other words in a semantic space. To construct this semantic space, it analyses word co-occurrences in a large number of written documents. Specifically, the semantic space used in all the examples below is based on a corpus of some 37,000 documents containing over 92,000 different word types - a total of about 11 million word tokens. From this statistical input LSA generates a high-dimensional semantic space by means of a mathematical technique called singular value decomposition, followed by dimension reduction. Thus, while the input to LSA consists of occurrence patterns over contexts, LSA does not represent meaning in terms of co-occurrence frequencies, but as vectors in a semantic space of 300-400 dimensions. The technique is related to factor analysis, but the dimensions of the space have no interpretation. The meaning of a word or sentence is represented by a vector of 300 numbers. This 300-dimensional space suffices to reconstruct not the accidental detail but the essential features of the original co-occurrence matrix and allows us to represent the meaning of arbitrary combinations of words and to compare them.

To find out whether an LSA vector correctly represents our semantic intuitions, we must compare it with other vectors. For instance, to determine whether the vector for a word means what it is supposed to mean, we can compare it with other words - landmarks - that we know to be related to it as well as with landmarks that we know are unrelated. We select these landmarks by our human intuition about the meaning of words and sentences; the question is whether LSA has the same kind of intuitions. Thus, we can compare the vector for the word *shark* with landmarks such as *fins*, *dolphin*, *diver*, and *fish* as well as some unrelated words. A quantitative measure of how close one vector is to another in the LSA space is given by the cosine between two vectors - a measure that can be interpreted in much the same way as a correlation coefficient. The cosines for *shark* and *fins*, *dolphin*, *diver*, and *fish* are .74, .74, .70, and .69, respectively. For comparison, the cosine between *shark* and *lawyer* is -. 01.[2]

LSA successfully captures one aspect of meaning - the semantic distance among words. Of course, LSA, like any scientific theory, is not the real thing – not meaning, but a model of meaning. Furthermore, it is an incomplete model. It models only those aspects of meaning that are coded verbally; human meaning is derived from perception and action as well as words. However, language has evolved to talk about perception and action, and one should not underestimate the power of the word to encode the human world. In addition, LSA has other limitations. For instance, it fails to explain the nature of the relation between *shark* and its neighbors – that is, how we understand that *a shark has fins, looks like a dolphin, is a danger to divers,* and *is a fish*. Neither does LSA distinguish *a shark is a fish* and *the fish is a shark*. Thus, LSA is not a complete model of meaning, but the fact that it allows us to compute automatically a quantitative measure of the relatedness between these terms is useful nevertheless. LSA can be an essential component of a psychological theory of meaning in that it provides a model of knowledge structure and a model of knowledge acquisition, based on tracking data about usage in the environment. But it needs to be combined with psychological process models of comprehension and thinking so as to achieve a full account of psychological semantics. In the present paper, LSA is paired with the CI model of text comprehension. This does not provide answers to all questions (e.g., it does not address the first of the limitations of LSA noted above – distinguishing between different types of relations) but

it does solve some problems (e.g., the second of the limitations discussed above – the asymmetry of arguments and predicates).

Vectors are the elements of an LSA semantics. The standard composition rule in LSA is the centroid rule, which says that the vector representing a set of words is the centroid of the individual word vectors. This rule is order insensitive. Nevertheless, in a large number of applications of LSA the centroid rule has proven to yield very useful results.[3] But the centroid rule is inadequate in many cases; one of the cases where the centroid rule fails is metaphorical predication. To use Glucksberg's example, if we compute the centroid of *lawyer* and *shark*, we land in a semantic no-man's land - somewhere in between *lawyer* and *shark*. Furthermore, composition by the centroid rule could not distinguish between *My lawyer is a shark* and *My shark is a lawyer*.

Kintsch (submitted) has argued that if we predicate something about a concept, not all the features of the predicate are combined with the meaning of the concept, but only those appropriate for that concept. Thus, different features of *run* play a role in *The horse runs* and The *color runs*. The argument - *horse*, or *color* - selects those features of the predicate that are appropriate for that argument, thus generating a contextualized word sense - the sense of *run* combined with *horse*, or the sense of *run* combined with *color*. That is all there is to metaphoric predication too: the argument selects those features of the (metaphoric) predicate that are appropriate for it and inhibits the features that do not apply or apply less aptly.

Consider the example from Glucksberg (1998), *My lawyer is a shark*. In his illustrative example Glucksberg lists nine features of *shark*, the first four of which are appropriate for the metaphor and enter its meaning, whereas the last five are irrelevant and are suppressed: *vicious, predatory, aggressive, tenacious, fast swimmer, fish, sharp teeth, leathery skin*, and *gills*. In fact, according to LSA, the last five, to-be-suppressed features are much more strongly related to *shark* (their average cosine with shark is .28) than the metaphor relevant features (their average cosine is .06), but when they are combined with *lawyer*, the typical shark features will be suppressed because they are unrelated to *lawyer* (their average cosine with *lawyer* is .01), and the atypical shark features will be emphasized because they are at least somewhat related to *lawyer* (their average cosine with lawyer is .08). The model proposed here provides a computational

algorithm that achieves this result. However, instead of describing the model in the context of Glucksberg's example where the predicate features to be considered were selected intuitively to make a point, the model will be described in its general case, which does not require an intuitive selection of features and is fully automatic.

The predication algorithm selects neighbors of a predicate that are related to the argument of the predication that are used to modify the predicate vector in order to make it context sensitive. It uses a spreading activation process in the manner of the construction-integration model to select among the terms in the LSA space that are related to P those that are also related to A, and then uses these terms to augment the vector representing the meaning of the metaphor. The general conceptual scheme will be described first, and then a computational approximation will be presented. The general scheme has the advantage that it makes clear just how the CI-model is combined here with LSA. The approximation does not employ the CI-model directly, but simplifies the computations significantly and yields equivalent results.

The predication algorithm first selects terms from the LSA space that are related to the predicate P, and then selects from this set those terms that are also related to A. The first step is achieved by computing the semantic neighborhood of P. The complete semantic neighborhood of a predicate P in the semantic space is a 300-dimensional hypersphere around P in which all 92,000 items in the semantic space are arranged according to their relationship with P. Items which have a high cosine with P will be near P, and items farther away are less and less related to P. In fact, most items will be at the periphery of the hypersphere centered on P, because they are essentially unrelated to P. One can order all the items in the space according to their cosine with P, generating a list of m words ordered in terms of their cosine with P.[4]

The second step involves constructing a spreading activation network in the manner of the construction-integration model. The network consists of A, P, and the m closest neighbors of P. Each term is connected to both P and A with a link strength corresponding to the cosine between the two nodes. In addition, each term is connected by an inhibitory link to every other term in the network. The strength of the inhibitory links are chosen in such a way that the total sum of all positive and negative links in the network is equal. If activation is spread in such a self-inhibitory network with the

activation values of P and A clamped at 1, most nodes will become deactivated and only those nodes related to both P and A will attain a positive activation value.

Finally, the k nodes with the highest activation values will be used to compute the vector representing the meaning of the metaphor. Specifically, the predication vector will be the centroid of A, P, and the k most highly activated terms of the network.

In actual computations, an approximation which greatly simplifies computations is employed for the second step described above. The sequence of steps in the computation of a predication vectors is therefore as follows:

1. Compute the semantic neighborhood of P of size m, as described above. For metaphors, m has to be fairly large ($500 < m < 1500$) because the predicate and argument in a metaphor often are quite unrelated.[5] This step assures that all terms that enter into the predication are in fact related to P.

2. The next step picks those terms from the neighborhood of P that are also related to A. The cosines between the m neighbors of P and A are computed and the k terms with the highest cosine are selected. This step obviates the need for setting up a huge self-inhibitory network and yields much the same results because there are usually only a few items related to both P and A and these would be selected in either case.

3. It is not necessarily the case that terms related to both P and A exist. Thus, in order to avoid introducing noise by selecting the strongest terms even when their absolute strength is low, the terms selected must have a cosine with P and A above some minimum threshold. Only terms that have a cosine with P greater than two standard deviations above the mean for all words in the space used here ($.02 + 2*(.06) = .14$) will be included among the to-be-considered items. Similarly, all terms related to A with a below-threshold cosine ($<.14$) will be eliminated.

4. The vector representing the meaning of the metaphor can then be computed as the centroid of A, and the terms selected above (P and the k terms from the neighborhood of P, subject to the restriction that their cosine with A is above threshold).

The centroid of A and B is the same as the centroid of B and A. Predication, in contrast, is basically asymmetric: if B is predicated about A, terms from the neighborhood of B that are compatible with A are used to modify the predication vector;

but if A is predicated about B, terms from the neighborhood of A are used in Step 1 of the procedure.

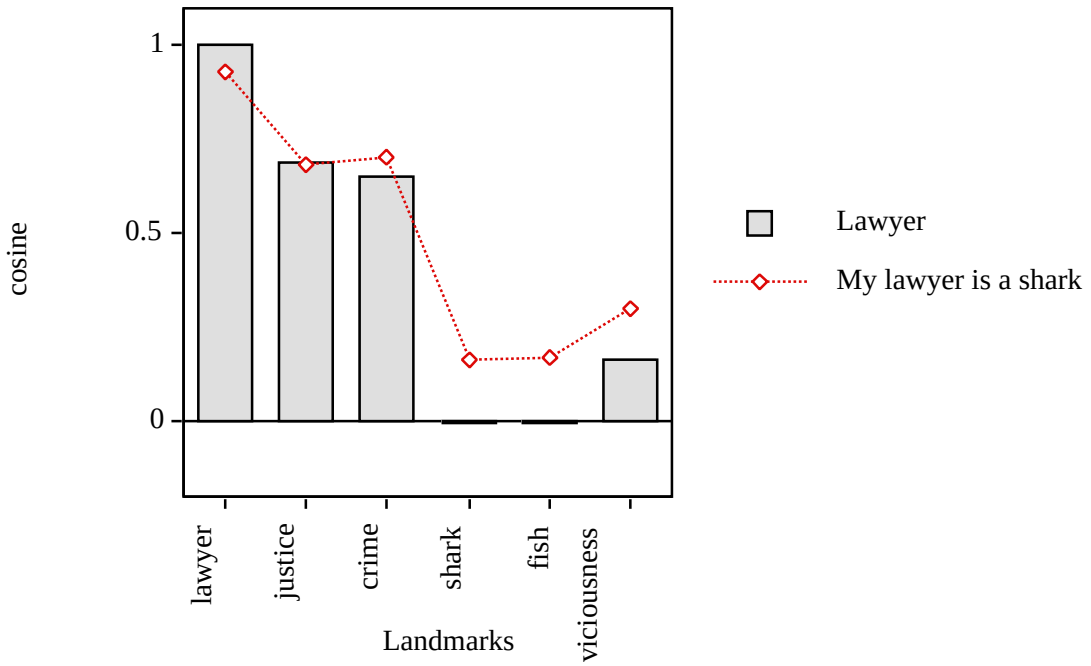

Figure 1. Vectors for *Lawyer* and *My lawyer is a shark* compared to six landmarks.

The predication algorithm yields a vector that needs to be interpreted by comparing it with suitable landmarks. In Figure 1 the vector for *My lawyer is a shark* is compared with six relevant landmarks. The vector was computed with k = 5 and m = 500. For these parameter values, the neighbors of P that were selected by the predication algorithm have an average cosine with P of .30 (range .27 - .35) and an average cosine with A of .20 (range .16 - .26). Thus, they are moderately strongly related to both P and A. The first three landmarks were chosen to be related to *lawyer* (the bars in Figure 1

show the magnitude of the cosine between each landmark and the single word *lawyer*); the second set of three landmarks was chosen to be related to *shark* - the first two items to the here inappropriate *fish*-sense of *shark*, and the third to be appropriate for the metaphor. Other landmarks similarly related to *lawyer* and *shark* could have been used. According to Figure 1, predicating *shark* about *lawyer* does not change the sentence meaning with respect to the *lawyer*-landmarks, but introduces a little fishiness and, primarily, moves the sentence meaning toward *viciousness*.

For lower values of m, the predication procedure fails: for m = 100, the meaning of *lawyer* is not modified at all because none of the 100 closest neighbors of *shark* have a cosine with *lawyer* that is greater than .14, the threshold value. For larger values of m (e.g., m = 1000 or 1250) essentially the same results are obtained as in Figure 1. For m = 1500, the algorithm begins to pick up too many *fish*-properties and the cosines with the landmarks *shark* and *fish* increase. At this point, the predication algorithm begins to converge with the centroid rule. The centroid of *lawyer* and *shark* behaves very different with respect to the landmarks in Figure 1 and clearly fails to represent the meaning of the metaphor: it is closer to the *shark* and *fish* landmarks (the cosines are .83 and .58, respectively) than either to *viciousness* or to the *lawyer*-landmarks[6].

In accordance with the claims of Glucksberg (1998), Kintsch (1998), and others, there is no difference in this theory between predication in the literal and metaphorical sense. For example, consider the literal statement *My lawyer is young*. The vector representing that sentence can be calculated with the same predication algorithm. For m = 50 and k = 5 the results shown in Figure 2 are obtained (they hardly change at all with changes in parameter values). Figure 2 uses the same landmarks for *lawyer* plus three new ones appropriate to the predicate. The results are interesting and contrast sharply with Figure 1. What we get is pretty much a straight combination of *lawyer* and *young* - there are no emergent features, no suppression, no surprises (indeed, the centroid of *lawyer* and *young* is not much different than the predication vector). When we say *My Lawyer is young*, we say little more than that person is young; none of the associated properties of *young* emerges as an important factor in determining the sentence meaning.

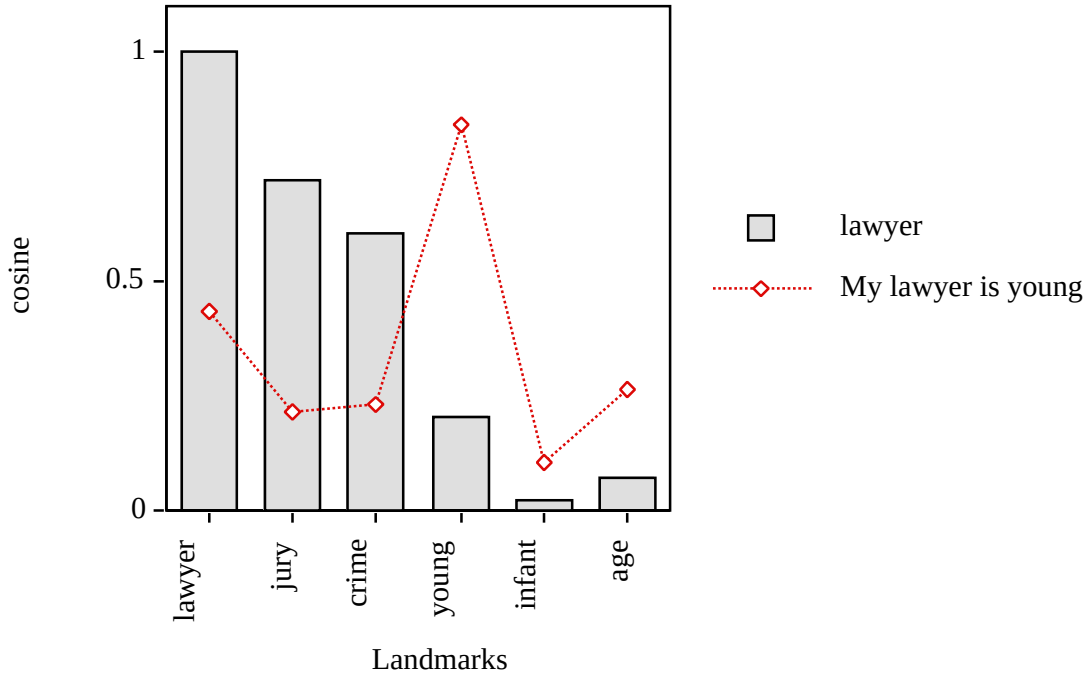

Figure 2. Vectors for *Lawyer* and *My lawyer is young* compared to six landmarks.

When is a sentence a metaphor, when is it a literal statement, and when is it just plain meaningless? Further research within the framework proposed here might yield some novel answers, but at present only a few hints can be offered here. Figures 1 and 2 illustrate one important difference between metaphors and literal statements: for the latter, argument and predicate are usually related in that many features of the predicate apply to the argument; the predicate selects and emphasizes one or more of these potential features of the argument. In metaphors, only a few features need to be related. In the case of *My lawyer is a shark*, topic and vehicle are not related at all by LSA (their cosine is -. 01). But for some metaphors, topic and metaphor can be related. For instance, of the 12 metaphors used in one esperimental study (Blasko & Connine, 1993), topic and vehicle were unrelated by LSA in only two cases, while for the other 10, the cosine between topic and vehicle was appreciable, ranging from .07 to .19. These were

metaphors such as *Rumors were plagues*, cos(*rumors, plagues*) = .15, or *The rocket was a bullet*, cos(*rocket, bullet*) = .16. These metaphors seem different in an important way from metaphors in which topic and vehicle are unrelated and seem more like literal statements. *Rumors were plagues* primarily attributes to *rumor* the feature *plagues*, plus some features associated with *plagues* (like *spreading*), much like a literal statement. Of course, not all features of *plagues* are attributed to *rumors* by the metaphor, but not all features of the predicate are attributed to an argument in literal predication either. For instance, *The color runs* is perfectly literal (it is listed as an example for one of the senses of *run* in WordNet), but only a judicious subset of *run*-features are attributed to *color* by this statement: we don't think that *the color runs like a machine, gallops like a horse, or moves through the tree like a breeze!* The question deserves to be explored more systematically, but it may be the case that for "real" metaphors, vehicle and topic are unrelated, whereas most of the Blasko and Connine examples might be regarded as intermediate forms between true metaphors and literal statements.

In order to assess the generality of the predication model, seven additional metaphors were analyzed. To avoid selection effects, the first seven examples of nominal metaphors cited in a well-known experimental paper (Glucksberg, Gildea & Bookin, 1982) were used for this analysis. However, the examples in Glucksberg et al. were all changed from plural forms *(Some salesmen are bulldozers*) to singular form (*The salesman is a bulldozer*) because many of the words involved were low frequency words and a preliminary analysis showed that LSA knew more about their singular forms than their plurals (e.g. *bulldozers* does not appear in the space used, but *bulldozer* does). The seven metaphors were *This job is a jail; Her marriage is an icebox; The salesman is a bulldozer; Her heart is a closet; The flute is a bird; The road is a snake;* and *My surgeon is a butcher*. The analysis was performed exactly in the same way as described above for the *My lawyer is a shark* example, that is with $\underline{m} = 500$, $\underline{k} = 5$.[7] Each vector computed with the predication procedure was compared with two landmarks, one relevant to the intended meaning of the metaphor, and one irrelevant to the metaphor but strongly related to another aspect of the predicate. Where possible these terms were selected from the dictionary definition of the predicate term.

The analysis yielded intuitively reasonable results in six out of the seven cases. The mean cosine between the predication vectors and the relevant landmarks was .36, whereas the mean cosine between the predication vector and the irrelevant landmark was .22 The *lawyer*-example shown in Figure 1 falls well within the range of these examples. In all six successful cases the cosine for the relevant landmark was greater than the cosine for the irrelevant landmark. This contrasted sharply with computations using the centroid of the argument and predicate as the vector representing the meaning of the metaphor: the cosines between the centroids and the relevant and irrelevant landmarks were approximately equal, .32 and .30, respectively. Thus, the predication algorithm selectively emphasizes appropriate semantic features of a metaphor, whereas the centroid imports relevant as well as irrelevant features of the predicate.

The predication procedure failed for the metaphor *Her marriage is an icebox*; the cosines between this metaphor and the relevant landmark *cold* and the irrelevant landmark *refrigerator* were both .03. There may be two reasons why predication failed in this case. First, LSA has very little information about *icebox* (vector length = .12, the lowest value in all examples), so that the neighborhood of *icebox* was rather vague and noisy. In addition, *marriage* is not related to *cold* and its synonyms in the LSA space used here, resulting in a failure of the selection mechanism. This lack of knowledge on the part of LSA is not totally surprising: the General Reading Space used here was constructed from a corpus consisting of the reading materials of an average high school student. It remains to be seen how well real high school students understand these metaphors. However, in future work, care must be taken to use words about which LSA is reasonably well informed; if the knowledge base is not there, the predication algorithm has nothing to work with.

The examples from Glucksberg et al. are, presumably, all examples of strong metaphors. What strong metaphors seem to have in common is that the predicate is a concrete term, rich in imagery and potential associations, and that the argument and predicate are relatively unrelated. The richness of the predicate allows the argument to resonate with several different features at the same time, resulting in a complex, if fuzzy, interpretation. The unrelatedness between the argument and predicate has surprise value. A strong the metaphor is something unusual, a pleasant surprise. But it cannot be too

much of a surprise. The semantic feature that was emphasized by the metaphor must already be inherent in the argument, even if at a low strength. In all cases were LSA yielded satisfactory interpretations, the argument and the relevant landmark were not completely unrelated. Thus, the effect of the predication was to emphasize some dormant but potential feature of the argument.

Experimental Findings on Metaphor Comprehension

The model proposed here not only can compute intuitively reasonable interpretations of metaphors (and literal statements), but it also provides an account for some of the major phenomena that have been studied in the experimental literature. Glucksberg (1998) serves as a good guide as to what these phenomena are.

1. Metaphors are in principle non-reversible. This is actually a claim that needs explanation. It really means two things:

(a) Some metaphors when reversed change their meaning. For example, *My surgeon is a butcher* and *My butcher is a surgeon* are both good metaphors but mean quite different things. This is not a problem for the present model, for in the one case properties of *butcher* are attributed to *surgeon*, and in the other properties of *surgeon* are attributed to *butcher*, as shown in Table 1.

Table 1. The cosines between *surgeon*, *butcher*, *My surgeon is a butcher* and *My butcher is a surgeon* and two landmarks, *scalpel* and *axe*.

	scalpel	**axe**
surgeon	.29	.05
butcher	.01	.37
My surgeon is a butcher	.10	.42
My butcher is a surgeon	.25	.26

(b) Some metaphors become meaningless when reversed. For instance, one can say *My job is a jail* but not **(My) jail is a job*. As always, however, the linguistic practice of starring sentences can be debatable; it is certainly possible to construct a scenario in which (*My) jail is a job* might be meaningful. It is obvious, however, that the original version of the metaphor is better than the reversed version. How can the model account for this?

Showing that something makes no sense is difficult. We can show that the metaphor in its original form does make sense: *My job is a jail* brings the sentence vector closer to *officer* and *lawyer*, which seems right intuitively. But *Jail is my job* emphasizes *hired* and *boss* - which at least to my intuitions isn't so bad either.

The theory is a laser beam highlights the *laser-beam* -properties *precision* and *light*. The reversed metaphor doesn't highlight typical *theory*-properties such as *explanation* and *hypothesis*. Similarly, *People are sheep* successfully transfers the *sheep*-property *follow* to the sentence vector, while *Sheep are people* does not import *people* properties such as *man*.

Thus, the model agrees with our intuitions about reversed metaphors, but offers no clear way to reject nonsensical sentences. In all examples above, there are terms in the neighborhood of the predicate that are related to the argument. In any case, it is obviously possible to predicate nonsense even about highly related words. Analyses that go beyond LSA and the CI-model may be needed at this point (as they are surely needed for other problems, too, e.g., the determination of what is a predicate and what is an argument in a sentence, which is a precondition for the predication algorithm, but outside the present scope of LSA).

2. <u>Bringing to mind the literal meaning of a metaphor vehicle has a deleterious effect</u>. To compute a metaphor vector, we construct a network out of the neighbors of the predicate P, which are linked to the argument A by their cosine values and inhibit each other. Initially, all nodes except A and P (the knowledge to be activated) have zero activation value, but activation flows into these nodes from A and P (the words of the

sentence that need to be interpreted). It requires several cycles of spreading activation in such a network before the activation values of the nodes stabilize. In isolation, My *lawyer is a shark* takes six iterations to settle. If *Sharks can swim* precedes the metaphor, the priming sentence will activate the neighborhood in a certain pattern, emphasizing the literal meaning of *shark*. That is, the neighbors of P will start with some positive activation value, depending on how strongly related they are to *sharks can swim.* Thus, the metaphor must be comprehended in the context of the priming sentence and the knowledge activation it produces.

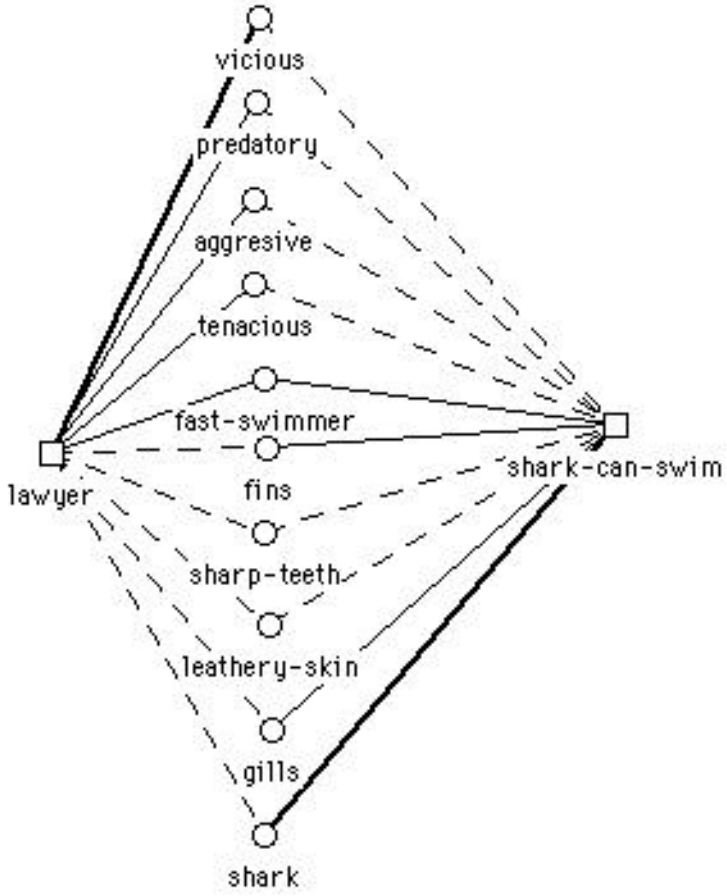

Figure 3. Properties of *shark* (after Glucksberg, 1998) related to the literal priming sentence *Sharks can swim* and to *lawyer*. Dashed lines indicate inhibitory connections, bold lines indicate strong links.

Instead of the whole set of neighbors of *shark*, the illustrative properties noted by Glucksberg (1998) are used in Figure 3 to keep the example simple. To integrate the above network the *lawyer* node is clamped at 1; *lawyer* and *shark-can-swim* are assigned a starting value of 1, whereas all other nodes have a starting values of 0; where links are shown in Figure 3, their strength is equal to the cosine between the respective nodes; in addition, there are links among all nine context nodes which were assigned a negative link strength in such as way that the absolute value of the total sum of the positive links equals the total sum of the negative links. Settling in this network requires 8 cycles, compared with 6 when the metaphor is understood out of context. That is not an impressive difference. But if one looks at the time course of integration, the experimental finding of slower comprehension with the prime becomes more understandable. In Figure 4 we see that if the metaphor is processed out of context, the *lawyer* relevant attributes dominate the integration process from the very beginning. In contrast, with the prime *sharks can swim*, the *shark*-specific attributes are stronger initially, and it takes several integration steps before this pattern is reversed. The final outcome is the same as without the prime, however. This agrees with the experimental findings of Glucksberg, Manfredini, and McGlone (1997) that people take more time to understand the primed metaphor, but arrive at the intended interpretation eventually.

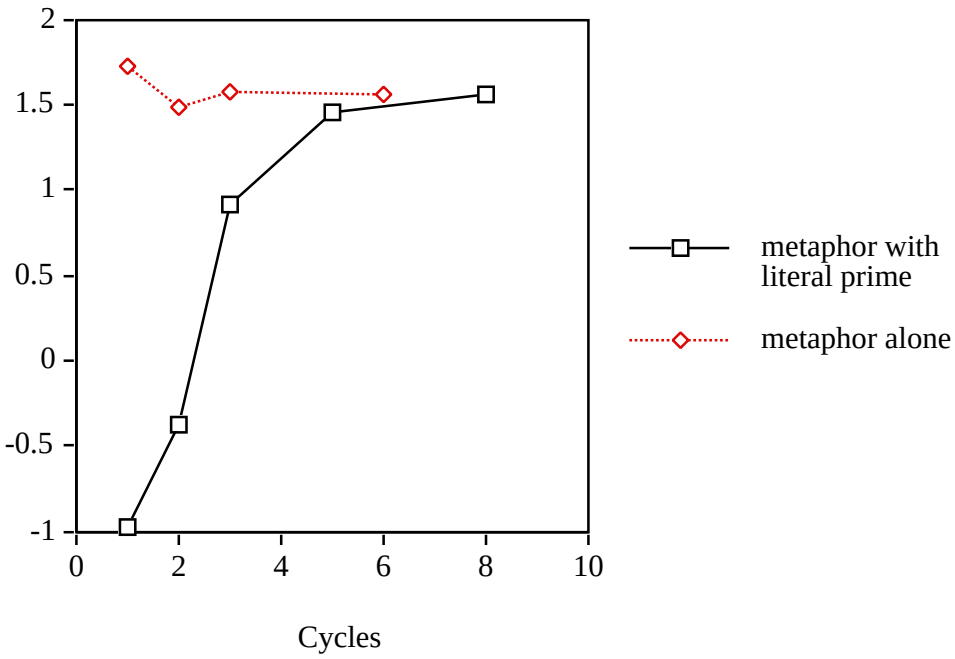

Figure 4. The difference in the sum of the activation values for appropriate and inappropriate properties of *My lawyer is a shark* in isolation and preceded by the literal prime *Sharks can swim* as a function of integration cycles.

3. Understanding a metaphor is like understanding any polysemous utterance. Compare *A rock fell off the mountain* and *The family is a rock.* The former is a literal statement, the latter is a metaphor. In both cases the meaning of *rock* must be computed in context, with quite different results. This is no different from computing one meaning of *bank* in the context of *money* and another meaning in the context of *river*. *Rock* (or any other word) takes on a slightly different meaning in each new context (Kintsch, 1998, Chapter 3).

Computing a meaning always involves activating context appropriate features and inhibiting or deactivating inappropriate features. Therefore, if some features have been

deactived and others strengthened in one context, and the context is changed, so that the deactivated features now become relevant and the activated features are irrelevant, it should take longer to form a stable meaning according to the CI-model than when no change in meaning is required. That is what was observed in an experiment by Gernsbacher, Keysar, and Robertson (1995). These authors have shown that literal statements are verified more slowly following a metaphor prime than following a literal prime. Their experimental design and their results are sketched below:

PRIME:	VERIFICATION STATEMENT:	REACTION TIME:
my lawyer is a shark	sharks are good swimmers	slow
the hammerhead is a shark	sharks are good swimmers	fast

The model predicts these results. The simulation is the reverse of the previous example. *Sharks are good swimmers* is clamped and must be interpreted either in the literal or metaphorical context. The results of the simulation are shown in Figure 5. In the context of the literal prime, *Sharks are good swimmers* requires 9 integration cycles to settle, versus 11 in the context of the metaphorical prime. However, Figure 5 shows that the metaphorical prime initially activates context irrelevant features, so that their activation is actually higher than the activation of relevant features. It requires several cycles before this interference is overcome. Eventually, of course, *Sharks are good swimmers* is understood correctly, but as Gernsbacher, Keysar, and Robertson (1995) observed, it takes more time to so.

Figure 5. The difference in the sum of the activation values for appropriate and inappropriate properties of *Sharks are good swimmers* preceded by the literal prime or metaphorical prime as a function of integration cycles.

Figure 5. The difference in the sum of the activation values for appropriate and inappropriate properties of *Sharks are good swimmers* preceded by the literal prime or metaphorical prime as a function of integration cycles.

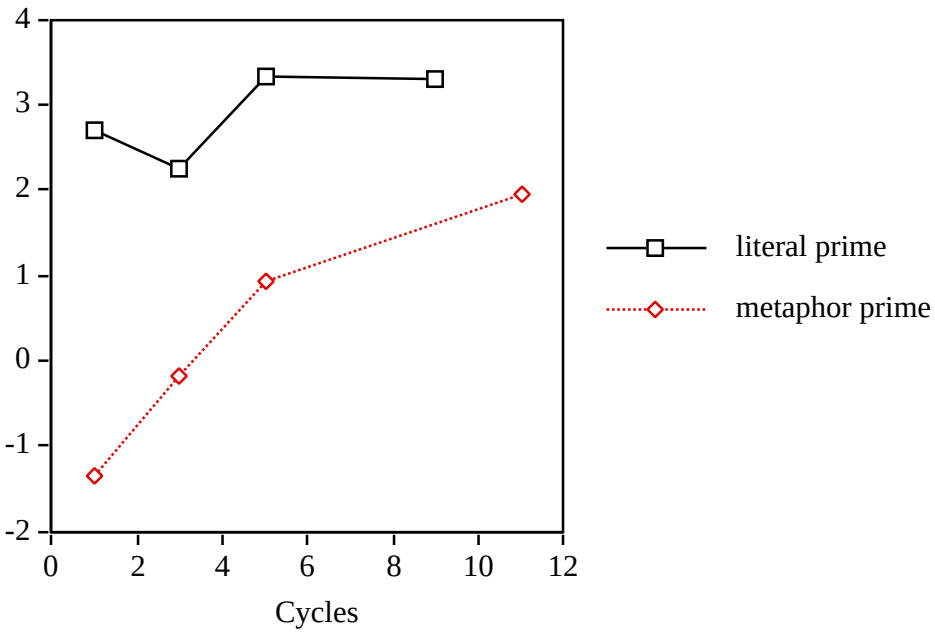

Figure 5. The difference in the sum of the activation values for appropriate and inappropriate properties of *Sharks are good swimmers* preceded by the literal prime or metaphorical prime as a function of integration cycles.

The Predication Algorithm and Theories of Metaphor

The most salient result of the experimental psycholinguistic research on metaphor has been the finding that metaphors are understood directly, much like literal statements - a result widely accepted today. The model proposed here embodies this premise. Indeed, the predication algorithm applies in the same way to literal and metaphorical predication. Several current theories of metaphor comprehension share this premise. For instance, this assumption is central to the theory of Glucksberg and Keysar (1990) and Glucksberg (1998). But Glucksberg's category inclusion theory of metaphors goes further than that, in that it postulates the creation of abstract categories for which the vehicle of the metaphor serves as a token (*shark* as a token for the category *predatory creatures*).

This may or may not be a good description of the predication algorithm proposed here: Glucksberg's theory requires a mechanism by means of which the topic of the metaphor is assigned to a newly created category. The vehicle of the metaphor names this category and also serves as the prototypical, defining member of that category. The predication algorithm is consistent with such a theory (if one accepts a very broad definition of the notion of category), but it certainly does not require it. Similarly, the present computational model is consistent with other theories of metaphor, without being dependent on them. For instance, Ortony's salience-imbalance theory (Ortony, 1979) defines metaphors in terms of particular relationships between topic and vehicle: a good metaphor is obtained when a property is associated with both the topic and the vehicle, but it is more salient in the vehicle, or when a term has low associations with both. There is nothing in the present model that restricts interpretations to these cases, but further research with the predication algorithm might show to what extent Ortony's claims can be substantiated.

It might seem that the present model is a member of the class of semantic feature models that treat metaphors as a comparison in the tradition of Aristotle and I. A. Richards (for a discussion, see Gibbs, 1994b). In this view, the inadequacy of which has been pointed out by Gibbs (1994b) and others, a feature associated with the vehicle is transferred to the topic. For instance, the feature *fierce* of *wolf* is transferred to *man* by the metaphor *Man is a wolf,* resulting in a meaning much like the literal statement *Man is fierce*. A problem with this view is that often there is no pre-existing association between the transferred feature and the vehicle. Gibbs (1994b) discusses the example *That girl is a lollipop,* which presumably means something like *That girl is frivolous. Frivolous,* however, is not a pre-existing association of *lollipop,* so there is nothing to transfer.

Gibb's criticism is to the point, but the present model is much more complex than the semantic feature theory he discusses. Indeed, it might be considered as a realization of the interactive theory of metaphor comprehension favored by modern scholars. It is not at all necessary that a feature emphasized by the metaphor be directly associated with the vehicle term. Indeed, this need not be the case: for *That girl is a lollipop* there is no pre-existing LSA relationship between the vehicle *lollipop* and *frivolous* (cos = .01), but nevertheless the meaning vector for the metaphor moves closer to *frivolous* (cos = .16)[8].

Why? Because the meaning vector is related to other terms which in turn are related to *frivolous* – e.g., to *friendly* (cos = .41), *smiled* (cos = .80), or *carnival* (cos = .39). The model thus does not pick out pre-existing associations, but rather merges two semantic neighborhoods. This merging is extremely selective and context sensitive, however, in that only the relevant terms are merged and the numerous irrelevant ones are suppressed. Somewhat related ideas are discussed by Gibbs (1994b) in terms of semantic fields. While LSA neighborhoods do not look much like the semantic fields linguists and philosophers have discussed (semantic neighborhoods are unstructured and not always intuitively interpretable), the analogy with semantic fields helps to differentiate the present model from the discredited semantic feature theory.

Thus, metaphors do not transfer a single feature, or even a small set of features, but rearrange a whole semantic field. This makes it difficult to evaluate some of the proposed theories of metaphor, such as Ortony's theory mentioned above, with the methods developed here. If *My lawyer is a shark* meant *My lawyer is vicious*, the task would be simple: we compute the cosine between *vicious* and both topic and vehicle, and see whether the relationship Ortony proposed holds.[9] But the meaning of a metaphor involves a restructuring of the semantic space, which is more difficult to capture than simple feature transfer.

No claim is made that the mechanism of the present model is the only one involved in metaphor comprehension. There are metaphors that demand more controlled analysis, especially literary metaphors, for example, in terms of analogical reasoning. Indirect comprehension of metaphors must certainly be possible: people can, and sometimes do, speculate about the meaning of a metaphor. It is also possible that judgements of the aptness of metaphors might involve processes other than those involved in comprehension. There are no reasons why the present model would be incompatible with additional processes that might also play a role in metaphor comprehension.

Conclusions

The predication model of metaphor comprehension described here has three components. First, LSA provides a model of human knowledge that is objective and

quantitative and can be used as the basis for a computational theory. Second, the CI theory is a suitable cognitive architecture for modeling the dynamics of comprehension. It allows us to adapt the general, context independent knowledge space of LSA to a particular context, in effect selecting from a large number of potential features of the vehicle precisely those that apply to the topic. Third, it offers a specific model of metaphor comprehension, by assuming that metaphoric predication works just like literal predication. None of these three components is new. LSA has been used to model human knowledge before (Landauer & Dumais, 1997); the CI architecture has provided the framework for a number of successful models of comprehension processes (Kintsch, 1998); and the claim that literal and metaphoric predication are alike has been supported by a number of researchers (e.g., Kintsch 1998, Chapter 3; Glucksberg, 1998; for more detail see the review articles cited earlier). What is new here is how these three components have been conjoined into a computational theory of metaphor comprehension that yields intuitively reasonable interpretations of metaphors and that accounts qualitatively for some of the major experimental results that have been obtained in this field.

As important as these results on metaphor comprehension are, it should not be overlooked that what has been proposed here is a general computational theory of predication in the LSA/CI framework. The early work on the CI model is entirely based on hand coding of propositions and the model had no objective way of modeling knowledge activation. LSA by itself does not distinguish between the roles of vehicle and topic, nor predicate and argument. In the present model, however, *A is a B* and *B is an A* are no longer (necessarily) the same. To explore the full implications of this model for predication is beyond the scope of this paper (but see Kintsch, submitted). Nevertheless, by showing how metaphor comprehension can be modeled, a further step has been taken towards the goal of an LSA-based computational model of language processing. The ability of LSA to represent human knowledge on a large scale provides exciting possibilities that need to be exploited.

References

Blasko, D. G., & Connine, C. M. (1993). Effect of familiarity and aptness on metaphor processing. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *19*, 295-308.

Cacciari, C., & Glucksberg, S. (1994). Understanding figurative language. In M. A. Gernsbacher (Ed.), *Handbook of psycholinguistics* (pp. 447-478). San Diego: Academic Press.

Foltz, P. W., Kintsch, W., & Landauer, T. K. (1998) The measurement of textual coherence with Latent Semantic Analysis. *Discourse Processes*, 25, 285-307.

Gernsbacher, M. A., & Faust, M. E. (1991). The mechanism of suppression: A component of general comprehension. *Journal of Experimental Psychology; Learning, Memory, and Cognition.*, *17*, 245-262.

Gernsbacher, M. A., Keysar, B., & Robertson, R. W. (1995). *The role of suppression in metaphor interpretation.* Paper presented at the annual meeting of the Psychonomic Society, Los Angeles.

Gibbs, R. W. Jr. (1994a). Figurative thought and figurative language. In M. A. Gernsbacher (Ed.), *Handbook of psycholinguistics* (pp. 411-446). San Diego: Academic Press.

Gibbs, R. W. Jr. (1994b*) The poetics of mind: Figurative thought, language, and understanding*. New York, NY: Cambridge University Press.

Glucksberg, S. (1998). Understanding metaphors. *Current Directions in Psychological Science*, *7*, 39-43.

Glucksberg, S., Gildea, P., & Bookin, H. B. (1982) On understanding non-literal speech: Can people ignore metaphors? *Journal of Verbal Learning and Verbal Behavior*, 21, 85-98.

Glucksberg, S & Keysar, B (1990) Understanding metaphorical comparisons: Beyond similarity. *Psychological Review*, 97, 3-18.

Glucksberg, S., McGlone, M. S., & Manfredini, D. A. (1997). Property attribution in metaphor comprehension. *Journal of Memory and Language, 36*, 50-67.

Kintsch, E., Steinhart , D., and the LSA Research Group (in press) Developing summarization skills through the use of LSA-based feedback. *Interactive Learning Environments.*

Kintsch, W. (1988). The use of knowledge in discourse processing: A construction-integration model. *Psychological Review, 95*, 163-182.

Kintsch, W. (1998). *Comprehension: A paradigm for cognition*. New York: Cambridge University Press.

Kintsch, W. (submitted) Predication.

Laham, D. (1997) Latent semantic analysis approaches to categorization. In M. G. Shafto & M. K. Johnson (Eds.), Proceedings of the 19th Annual Conference of the Cognitive Science Society (p.979) Mahwah, NJ: Erlbaum.

Laham, D., Bennett, W. Jr., & Landauer, T. K. (in press). An LSA-based software tool for matching jobs, people, and instruction. *Interactive Learning Environments.*

Landauer, T. K. & Dumais, S. T. (1997). A solution to Plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. *Psychological Review*, 104, 211-240.

Landauer, T. K., Laham, D., Rehder, B., & Schreiner, M. E., (1997). How well can passage meaning be derived without using word order? A comparison of Latent Semantic Analysis and humans. In M. G. Shafto & P. Langley (Eds.), Proceedings of the 19th annual meeting of the Cognitive Science Society (pp. 412-417). Mawhwah, NJ: Erlbaum.

Landauer, T. K., Foltz, P., & Laham, D. (1998). An introduction to Latent Semantic Analysis. *Discourse Processes*, 25, 259-284.

Ortony, A. (1979). Beyond literal similarity. *Psychological Review*, 86, 1671-180.

Wolfe, M. B., Schreiner, M. E., Rehder, R., Laham, D., Foltz, P. W., Landauer, T. K., & Kintsch, W. (1998). Learning from text: Matching reader and text by Latent Semantic Analysis. *Discourse Processes*, 25, 309-336.

Wolfe, M. B., Schreiner, M. E., Rehder, R., Laham, D., Foltz, P. W., Landauer, T. K., & Kintsch, W. (1998). Learning from text: Matching reader and text by Latent Semantic Analysis. *Discourse Processes*, 25, 309-336.

Footnotes

This research was supported by grants from the Army Research Institute and the J. S. McDonnell Foundation. I thank Tom Landauer, Eileen Kintsch, Dave Steinhart and the other members of the Colorado LSA Group for their help, comments, and simulating discussions.

---

[1] The discussion in this paper is restricted to attributional metaphors of the form "A is P," where A is the topic of the metaphor (the argument of the underlying proposition) and P is the vehicle of the metaphor (the predicate of the proposition). The theory presented is a general one, however, and the extension to other forms of simple predication is straightforward.

[2] All computations are based on the General Reading Space with 300 dimensions and can be performed at the LSA web site, http://lsa.colorado.edu. In general, only the relative values of cosines are readily interpretable, but not their absolute values.

[3] LSA has proven to be a powerful tool for modeling psychological phenomena such as simulating the rapid acquisition of vocabulary (Landauer & Dumais, 1997), categorization (Laham, 1997), the analysis of textual coherence (Foltz, Kintsch, & Landauer, 1998) and practical applications requiring the representation of meaning, such as essay grading (Landauer et al.,1997), helping students to write summaries (E. Kintsch et al., in press), selecting instructional materials suitable for a student's background knowledge (Wolfe et al., 1998), and selecting personnel with the knowledge required for specific jobs (Laham et al., in press).

[4] The Nearest-Neighbor/term program available at the LSA web site does exactly that.

[5] For literal sentences, much smaller values of m are sufficient, e.g. m = 20 (Kintsch, submitted).

[6] The centroid of *lawyer* and *shark* reflects *shark*-properties more strongly than *lawyer* properties because the length of the *shark* vector is greater than the length of the *lawyer*

vector (.87 versus .57, respectively). That is, LSA knows more about *shark* than about *lawyer*, and this greater knowledge biases the average in favor of *shark*.

[7] The General Reading Space includes some very rare words, as well as a few misspellings and word fragments. Only words that could be found in the American Heritage Dictionary were included in the analysis.

[8] Calculations are based on a semantic neighborhood of *lollipop* of size 50, which yielded 23 terms related to *girl*.

[9] In fact, *vicious* is more closely related by the cosine measure to *lawyer* than to *shark* - but that may be an idiosyncrasy of the General-Reading space used here for the LSA analysis, which is trained more on biology texts than on horror stories.