Law, K., Forbus, K. D., & Gentner, D. (1994). Simulating similarity-based retrieval: A comparison of ARCS and MAC/FAC. Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society, 543-548. Hillsdale, NJ: Lawrence Erlbaum Associates.

Simulating Similarity-Based Retrieval: A Comparison of ARCS and MAC/FAC

Keith Law
Kenneth D. Forbus
The Institute for the Learning Sciences
Northwestern University
1890 Maple Avenue
Evanston, IL, 60201
law@ils.nwu.edu
forbus@ils.nwu.edu

Dedre Gentner
Psychology Department
Northwestern University
2029 Sheridan Road
Evanston, IL 60208
gentner@nwu.edu

Abstract

Current theories and supporting simulations of similarity-based retrieval disagree in their process model of semantic similarity decisions. We compare two current computational simulations of similarity-based retrieval, MAC/FAC and ARCS, with particular attention to the semantic similarity models used in each. Four experiments are presented comparing the performance of these simulations on a common set of representations. The results suggest that MAC/FAC, with its identicality-based constraint on semantic similarity, provides a better account of retrieval than ARCS, with its similarity-table based model

1. Introduction

How does a pendulum remind us of a spring, or even of another pendulum? This paper compares two recent simulations of how such remindings come about: ARCS (Thagard, Holyoak, Nelson & Gochfeld, 1990) and MAC/FAC (Gentner, 1989; Gentner & Forbus, 1991, in preparation; Gentner, Rattermann & Forbus, 1993). Both models attempt to predict the fact that similarity-based retrieval is strongly influenced by surface similarity and weakly sensitive to structural consistency. The process should typically retrieve literally similar matches, often retrieve surface-similar matches, and occasionally retrieve purely analogous matches (Gentner, Rattermann & Forbus, 1993; Gick & Holyoak, 1980, 1983; Wharton, Holyoak, Downing, Lange and Wickens, 1991, in preparation).

Section 2 reviews MAC/FAC and ARCS. Section 3 describes four computational experiments in which we compare MAC/FAC and ARCS. Section 4 summarizes the results.

2. Review of MAC/FAC and ARCS

MAC/FAC: MAC/FAC (for "Many are called but few are chosen") uses a two-stage retrieval process. The first stage (MAC) is a "wide-net" stage in which a crude, computationally cheap, match process is used to pare down the vast set of memory items into a small set of candidates for more expensive processing. The second stage (FAC) uses SME in literal similarity mode to apply structural constraints to select one (or a few) best matches.

Figure 1 summarizes the MAC/FAC algorithm. The MAC stage operates with *content vectors*, a vector representation automatically computed from structured representations. Each component of a content vector represents the relative number occurrences of a particular predicate in the corresponding structured representation. Thus the dot product of two content vectors yields an estimate of how likely their corresponding structured representations will match using SME. Given a probe, its content vector is computed and its dot product taken with every item in memory. The output of the MAC stage is the item with the highest dot product, along with everything else within 10% of it.

The FAC stage uses SME to calculate, in parallel, a structural alignment of each item retrieved by MAC with the probe. Since MAC is sensitive only to predicate overlap while FAC is sensitive to structure, FAC will reject much of MAC's output. However, MAC's pre-filtering minimizes the number of structural alignments to be computed.

ARCS: The ARCS algorithm is shown in Figure 2. ARCS uses a localist connectionist network to apply semantic, structural, and pragmatic constraints to selecting items from memory. The initial stage uses semantic similarity to select a subset of memory over which to build a matching network. The notion of semantic similarity is

Given a database M of memory items I1..In, and a probe P,

1. *[MAC stage]* In parallel, for each item I in M compute the dot product of the content vectors for I and P. Return as output the maximum and every item whose score is within *p1%* of it.
2. *[FAC stage]* In parallel, for each item I in the MAC output, run SME with I as the base and P as the target. The FAC score for each pair is the structural evaluation score of the highest- ranked mapping. The top-scoring match, plus any others within *p2%* of it, are output.

(Typically p1 = p2 = 10%)

Figure 1: The MAC/FAC algorithm

Given a pool of memory items I1..In and a probe P:

1. For each item Ii, include it in a matching network if there are any predicates in Ii that are semantically similar to a predicate in P. The matching network implements semantic and structural constraints.
2. Create inhibitory links between units representing competing retrieval hypotheses, to ensure competitive retrieval.
3. Install pragmatic constraints by creating excitatory links between a special pragmatic node and every predicate marked by the user as important.
4. Run the network until it settles.

Figure 2: The ARCS algorithm

based on WordNet (Miller, Fellbaum, Kegl, & Miller, 1988), a psycholinguistic database of words and lexical concepts. Since Thagard et al. draw the majority of their predicate vocabulary from WordNet, the existence of lexical relationships between words is used to suggest that their corresponding predicates are semantically similar.

Most of the work in ARCS is carried out by the constraint satisfaction network, which provides an elegant mechanism for integrating the disparate constraints that Thagard et al. postulate as important to retrieval. The use of competition in retrieval is designed to reduce the number of candidates retrieved. Using pragmatic information provides a means for the system's goals to affect the retrieval process.

After the network settles, an ordering can be placed on nodes representing retrieval hypotheses based on their activation. Unfortunately, we have not been able to identify a formal criterion by which a subset of these retrieval hypotheses are considered to be what is retrieved by ARCS. In the experiments below we mainly focus on the subset of retrieval nodes mentioned by Thagard et al. in their paper.

2.1 Semantic Similarity

A key issue in analogical processing is what criterion should be used to decide if two elements can be placed into correspondence. In ARCS, an augmented subset of WordNet was used to make semantic similarity decisions. Two predicates in ARCS are considered semantically similar if their corresponding lexical concepts in WordNet are connected via links that denote particular relationships. The use of WordNet as a database for simple lexical inferences is an appealing idea. The lexical connections found in this way should have well-founded motivations. Nevertheless, it important to remember that WordNet was intended as a lexicon, not a language of thought. Using the lexical concepts of WordNet as a predicate vocabulary requires assuming that there exist conceptual representations that correspond to these lexical concepts. That does not seem an implausible assumption. However, assuming that relationships between words, such as *synonym* or *antonym* are used in the cognitive processing of internal representations seems implausible.

We prefer an identicality-based account using inexpensive inference techniques to suggest ways to re-represent non-identical relations into a canonical representation language. Such canonicalization has many advantages for complex, rich knowledge systems, where meaning arises from the axioms that predicates participate in. When mismatches occur in a context where it is desirable to make the match, we assume that people make use of techniques of re-representation. An example of an inexpensive inference technique to suggest re-representation is Falkenhainer's (1987, 1990) *minimal ascension* method, which looks for common superordinates (e.g., TRANSFER) when context suggested that two predicates should match (e.g., BESTOW and DONATE). Semantic similarity can thus be captured as partial identity. We believe that WordNet could be used similarly, since it has superordinate information.

Holyoak & Thagard have argued that broader (i.e., weaker) notions of semantic similarity are crucial in retrieval, for otherwise we would suffer from too many missed retrievals. Although this at first sounds reasonable, there is a counter-argument based on memory size. Human memories are far larger than any cognitive simulation yet constructed. In such a case, the problem of false positives (i.e., too many irrelevant retrievals) becomes critical. False negatives are of course a problem, but they can be overcome to some extent by reformulating and re-representing the probe, treating memory access as an iterative process interleaved with other forms of reasoning (as in Wharton, Holyoak, Downing, Lange & Wickens's (1991, in press) REMIND model). Thus we argue that strong semantic similarity constraints, combined with re-representation, are crucial in retrieval as well as in mapping.

How do these different accounts of semantic similarity fare in predicting patterns of retrieval? In the rest of the paper we compare the performance of MAC/FAC and ARCS on a variety of examples.

3. Computational Experiments

Each experiment below has a similar structure. First each simulation is given a memory, consisting of one or more databases drawn from the ARCS representations.[1] Then retrieval is tested with probes drawn from a small predefined set of stories. The memory a simulation operates over consists of one or more databases. In some cases the memory is augmented by a particular story: e.g., when probing with variant Hawk stories, the Thagard et al. encoding of the "Karla the Hawk" story is added to memory. (This is done to see if the retrieval system is able to find the base story amidst the distractors, given variations on the story as probes.)

For brevity we specify the probe set and memory contents symbolically, using "/" to distinguish probe set from memory and "+" to indicate set union. Thus

[1] To date we have been unsuccessful in getting ARCS to run on the representations we used in (Forbus & Gentner, 1991). ARCS' network does not settle after even 1,000 iterations, and run times of up to nine hours have been required.

HAWK/(PLAYS+Karla Base) indicates an experiment where the database of plays was probed with the Hawk stories. A description of the datasets used and a summary of conventions are given in Figure 3.

Both MAC/FAC and ARCS take propositional representations as inputs, but their representation conventions are quite different. The most crucial difference is that structure-mapping treats attributes, relations, and functions differently, whereas ARCS does not distinguish them. We used the following rules in translation: (1) One-place predicates were classified as attributes, (2) multi-argument predicates were classified as relations, and (3) since the arguments to CAUSE could be either events or modal propositions, we treated predicates used as arguments to a CAUSE statement either as modal relations (e.g., BECOMING-TRUE) or functions (e.g., MARRIED, KILLED).

Replication of computational experiments is still something of a novelty, and standards for ensuring that reported simulation results are repeatable have not yet been established in cognitive science. Nevertheless, we have taken many precautions to ensure that we have run ARCS correctly. Where numerical information was reported, for instance, we matched results to several decimal places. One concern was what should count as a retrieval in ARCS. Neither the original ARCS paper nor the code defines a criterion for distinguishing when an item is actually retrieved (indeed, stories with negative activations were sometimes considered retrievals). In reporting ARCS results we cut off the list of retrieved results where they did. In some cases (e.g., fables) this represented a sharp boundary, in other cases (e.g., plays) it did not.

Databases:
FABLES = 100 encodings of Aesop's fables, encoded by Thagard et al.
PLAYS = 25 encodings of Shakespeare's plays, encoded by Thagard et al.
Story sets used as probes and memory items:
HAWK = Thagard et al.'s encoding of the "Karla the Hawk" story set, i.e., original story, analog, appearance match, false analogy, and literal similarity versions. Databases using these probes have the original story added to memory, except when the original story itself is used as a probe.
SG = Thagard et al.'s encoding of the Sour Grapes fable plus variations, i.e., original story, analog, appearance, and literal similarity versions. Databases using these probes have the original story added to memory, except when the original story itself is used as a probe.
H&WSS = Thagard et al.'s encoding of Hamlet and West Side Story. When Hamlet is used as a probe it is removed from memory. West Side Story is never placed in memory.
Convention: For convenience, we refer to an experimental setup by the probe stories followed by the database used, e.g., SG/(FABLES+PLAYS) means that the Sour Grapes fables were used as probes with a memory consisting of both plays and fables. When a story is used as a probe, it is removed from memory first.

Figure 3: Databases and experimental stories used in the experiments

ARCS results. Numbers in parentheses represent the level of activation computed by ARCS.

Probe	Results	Sec
Sour Grapes, appearance	Sour Grapes (0.28)	120
Sour Grapes, analog	Sour Grapes (0.21)	81
Sour Grapes, literal similarity	Sour Grapes (0.25)	123

MAC/FAC Results. Numbers in parentheses represent the scores for that story.

Probe	Results	Sec
Sour Grapes, appearance	FAC: Sour Grapes (0.53) MAC: Sour Grapes (0.56)	0.3
Sour Grapes, analog	FAC: Sour Grapes (2.03) MAC: Sour Grapes (0.62)	0.2
Sour Grapes, literal similarity	FAC: Sour Grapes (2.03) MAC: Sour Grapes (0.62)	0.2

Table 1: Results for SG/Fables experiment

3.1 Experiment 1: Sour Grapes Comparison

In the first study the memory set consists of the fables, including the Sour Grapes fable, and the probes are variants of Sour Grapes. Table 1 shows the results. The results for ARCS match those reported for the simulation by Thagard et al. The MAC/FAC results are quite similar. Thus both systems successfully retrieve Sour Grapes from a database of fables when given variations of it. However, MAC/FAC is substantially faster. The runtime difference is fairly typical; MAC/FAC tends to be two orders of magnitude faster than ARCS when tested with identical data on the same computer.

3.2 Experiment 2: Effects of additional memory items on retrieval (Sour Grapes)

To check the stability of results under changes in memory contents, we reran Experiment 1, adding the database of 25 Shakespeare plays encoded by Thagard et al. to the fables database. We then tested the simulations to see if they would retrieve Sour Grapes from the database of 125 fables and plays when probed with variations of Sour Grapes. The results are show in Table 2. MAC/FAC's results remain unchanged, except for a small increase in processing time. ARCS, on the other hand, is distracted by the plays in one of the probe conditions. Increasing the memory by 25% has led to different results with ARCS. The results also hint at a possible size bias in ARCS; it appears to prefer larger descriptions in retrieval, at the cost of correct matches.

ARCS Results		
Probe	**Results**	**Sec**
Sour Grapes appearance	Sour Grapes (0.28)	327
Sour Grapes, analog	The Taming of the Shrew (0.22), Merry Wives (0.18), [11 stories], Sour Grapes (-0.19)	251
Sour Grapes, literal similarity	Sour Grapes (0.25)	373
MAC/FAC Results		
Probe	**Results**	**Sec**
Sour Grapes appearance	FAC: Sour Grapes (0.53) MAC: Sour Grapes (0.56)	0.4
Sour Grapes analog	FAC: Sour Grapes (2.03) MAC: Sour Grapes (0.62)	0.3
Sour Grapes, literal similarity	FAC: Sour Grapes (2.03) MAC: Sour Grapes (0.62)	0.3

Table 2: Results of SG probes, database = Fables + Plays

3.3 Experiment 3: Larger Probe sizes

While the results for MAC/FAC in Experiment 2 are satisfactory, ARCS' seemingly poor performance requires further investigation. Does the relative size of the probe matter in the memory swamping effect? To find this out, we again ran both simulations, first with the plays database as memory, then with the 25 plays and 100 fables as memory, this time using as probes the Hamlet and West Side Story encodings as probes, as represented by Thagard et al. Given Hamlet as a probe, the question is whether the systems can retrieve a tragedy, or at least another play. Given West Side Story as a probe, the challenge is more specific: to retrieve Romeo & Juliet, the analogous play.

Table 3 shows the results for plays only in memory, and Table 4 shows the results with both plays and fables in memory. The good news for ARCS is that the fables have only minimally intruded on the activation for the top ranked retrieved plays. A Midsummer Night's dream is ARCS' top-ranked retrieval for West Side Story, but it did also, as stated by Thagard et al., retrieve Romeo & Juliet.

MAC/FAC, on the other hand, only retrieves Romeo & Juliet with either probe. For West Side Story this is indeed the expected result (and we believe more intuitive that ARCS' result), but what is happening with Hamlet? Examining the structural evaluation scores (e.g., the FAC scores) reveals that FAC considers the match between West Side Story and Romeo & Juliet to be excellent (16.51), which makes sense because the encodings of West Side Story and Romeo & Juliet have almost isomorphic structure. When Hamlet is the probe, FAC is relatively indifferent: the FAC scores were as follows: Romeo & Juliet (6.79), Julius Caesar (5.49), Macbeth (3.72), Othello (2.67). The drop-off from Romeo & Juliet is 20%, which is below than MAC/FAC's default cutoff of 10%.

3.4 Experiment 4: Hawk stories

The goal in the Hawk studies was to replicate the results of (Gentner, Rattermann, & Forbus, 1993). Subjects were given a set of stories to read, and later attempted to retrieve these stories given variations as probes. The observed retrieval ordering was literal similarity, appearance, analogy, first-order overlap. Thagard et al. simulated this experiment for one story set. Using the relative activation levels of the stories computed by ARCS as relative retrieval probabilities for human subjects, ARCS' order of retrieval was: literal similarity, first-order overlap, appearance, analogy. This is not a close match. (Our own simulation of these results with MAC/FAC matched the human ordinal results.)

ARCS results. Numbers in parentheses represent levels of activation for that item.		
Probe	**Results**	**Sec**
Hamlet	Romeo & Juliet (0.54), King Lear (0.53), Othello (0.46), Cymbeline (0.42), Macbeth (0.41), Julius Caesar (0.38)	1843
West Side Story	Midsummer Night's Dream (0.58), Romeo & Juliet (0.57)	2539
MAC/FAC results. Numbers in parentheses represent scores for that item.		
Probe	**Results**	**Sec**
Hamlet	FAC: Romeo & Juliet (6.79) MAC: Othello (0.86), Macbeth (0.85), Romeo & Juliet (0.83), Julius Caesar (0.81)	22
West Side Story	FAC: Romeo & Juliet (16.51) MAC: Romeo & Juliet (0.88)	13

Table 3: Results for Hamlet, West Side Story as probes, Plays database.

ARCS Results.		
Probe	**Results**	**Sec**
Hamlet	Romeo & Juliet (0.531), King Lear (0.528), Othello (0.45), Cymbeline (0.41), Macbeth (0.40), Julius Caesar (0.37)	4112
West Side Story	Midsummer Night's Dream (0.58), Romeo & Juliet (0.57)	5133
MAC/FAC Results		
Probe	**Results**	**Sec**
Hamlet	FAC: Romeo & Juliet (6.79) MAC: Othello (0.86), Macbeth (0.85), Romeo & Juliet (0.83), Caesar (0.81), Fable52 (0.80)	26
West Side Story	FAC: Romeo & Juliet (16.51) MAC: Romeo & Juliet (0.88)	8

Table 4: Results for Hamlet, West Side Story as probes, Plays + Fables database.

However, our purpose here is to pursue two specific questions. Using Thagard et al.'s encodings, we ask (1) do the systems perform appropriately; and (2) do the two systems continue to perform appropriately when distractors are added to memory? Both simulations were run with the Hawk stories as probes, and with either the fables (plus the Karla story) as memory or with both fables and plays (plus the Karla story) as memory. The results are shown in Table 5 and Table 6 respectively.

No matter which database is used, MAC/FAC always retrieves the Karla story, irrespective of which variant story is used as a probe. The MAC scores explain why: In each case the Karla story is at the top of the ranking, indicating that the predicate overlap is greater for Karla and variant than for any other story. The fact that the Karla base story is retrieved for the literal similarity and appearance variants is expected. Its retrieval when the analogy is used as a probe is also reasonable (although if ARCS always retrieved analogs successfully it would be an implausible model). Retrieving the base story when the first-order overlap story is used as a probe is not so reasonable. We believe this occurs because the Thagard et al. representations are rather sparse, with almost no surface information, and thus are less natural than might be desired (c.f. the *specificity conjecture* of Forbus & Gentner, 1989).

As was suggested by experiments 1 and 2, the ARCS results vary considerably with different distractor sets. This means that the use of relative activations to estimate relative frequencies is not a stable measure. Specifically, the relative ordering of first-order overlap and analogy reverses when the database of fables is augmented with the plays. The position of the Karla story in the activation rankings is also alarming. The appearance story, which

ARCS Results

Probe	Results	Sec
Karla, literal similarity	"Karla" base (0.67)	315
Karla, appearance	Fable55 (0.4), [7 fables], "Karla" base (-0.17)	176
Karla, analogy	Fable23 (0.33), [7 fables], "Karla" base (-0.27)	127
Karla, first-order overlap	Fable23 (0.0907), Fable55 (0.0903), [13 fables], "Karla" base (-0.11)	17

MAC/FAC Results.

Probe	Results	Sec
Karla, literal similarity	FAC: "Karla" (16.07) MAC: "Karla" (0.81), Fable71 (0.74)	6
Karla, appearance	FAC: "Karla" (7.92) MAC: "Karla" (0.71), Fable52 (0.71), Fable71(0.66), Fable27(0.65), Fable5(0.64)	7
Karla, analogy	FAC: "Karla" (8.57) MAC: "Karla"(0.81), Fable52 (0.77), Fable5 (0.77), Fable71(0.76), Fable45(0.75), Fable59(0.75), Fable27(0.75)	14
Karla, first-order overlap	FAC: "Karla" (5.33), Fable5 (5.33) MAC: "Karla" (0.73), Fable71(0.71), Fable52(0.71), Fable5(0.71), Fable45(0.69), Fable59(0.68),Fable27(0.68)	7

Table 5: Results for HAWK probes, database = Fables + "Karla" base story

ARCS Results.

Probe	Results	Sec
Karla, literal similarity	"Karla" base (0.67)	614
Karla, appearance	Fable55 (0.40),[16 stories], "Karla" base (-0.018)	408
Karla, analogy	Pericles (0.60), [17 stories], "Karla" base (-0.32)	244
Karla, first-order overlap	Pericles (0.58), [22 stories], "Karla" base (-0.38)	45

MAC/FAC Results.

Probe	Results	Sec
Karla, literal similarity	FAC: "Karla"(16.07) MAC: "Karla"(0.81), Fable71 (0.74)	7
Karla, appearance	FAC: "Karla" (7.92), MAC: "Karla" (0.71), Fable52(0.71), Julius Caesar (0.69), Othello (0.68), Macbeth (0.67), Fable71(0.66), Two Gentlemen of Verona (0.65), Fable27(0.65), Hamlet (0.65), Fable5(0.64)	21
Karla, analogy	FAC: "Karla"(8.57) MAC: "Karla" (0.81), Julius Caesar (0.78), Two Gentlemen of Verona (0.78), Fable52 (0.77), Fable5(0.77), Macbeth (0.76), As You Like It(0.76), Fable71(0.76), Fable45(0.75), Fable59(0.75), Fable27(0.75), Othello(0.75)	37
Karla, first-order overlap	FAC: "Karla"(5.33), Fable5(5.33), As You Like It (4.96) MAC: "Karla"(0.73), Julius Caesar(0.72), Two Gentlemen of Verona (0.72), Fable71(0.71), Fable52(0.71), Fable5 (0.71), Macbeth(0.70), As You Like It (0.70), Othello (0.69), Fable45 (0.69), Hamlet(0.68)	22.6

Table 6: Results for HAWK probes, with database = Fables + Plays + "Karla" base story

should retrieve the base almost as often as the literal similarity story, has dropped from ninth in the ranking to 18th. Depending on the retrieval cutoff, the conclusion might be that ARCS fails to retrieve the Karla story given the very close surface match.

4. Conclusions

The results of cognitive simulation experiments must always be interpreted with care. In this case, we believe our experiments provide evidence that structure-mapping's identicality constraint better models retrieval than Thagard et al.'s notion of semantic similarity. In retrieval, the special demands of large memories argue for simpler algorithms, simply because the cost of false positives is much higher. If retrieval were a one-shot operation, the cost of false negatives would be higher. But in normal situations, retrieval is iterative, interleaved with the construction of the representations being used. Thus the cost of false negatives is reduced by the chance that reformulation of the probe, due to re-representation and inference, will subsequently catch a relevant memory that slipped by once.

Finally, we note that while ARCS' use of a localist connectionist network to implement constraint satisfaction is in many ways intuitively appealing, it is by no means clear that such implementations are neurally plausible. Overall, we believe the evidence suggests that MAC/FAC captures similarity-based retrieval phenomena better than ARCS does.

5. Acknowledgments

We thank Paul Thagard and Keith Holyoak for providing ACME, ARCS, and advice on operating them. This research was funded by the Cognitive Science Division of the Office of Naval Research.

6. Bibliography

Falkenhainer, B. (1987). An examination of the third stage in the analogy process: Verification-based analogical learning. In *Proceedings of IJCAI-87*. Los Altos: Morgan Kaufmann.

Falkenhainer, B. (1990). A unified approach to explanation and theory formation. In Shrager, J. & Langley, P. (Eds.), *Computational Models of Scientific Discovery and Theory Formation*. Morgan Kaufmann.

Falkenhainer, B., Forbus, K., & Gentner, D. (1989). The Structure-Mapping Engine: Algorithm and Examples. *Artificial Intelligence*, 41, 1-63.

Forbus, K. & Gentner, D. (1989). Structural evaluation of analogies: What counts? In *Proceedings of the Eleventh Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Forbus, K. & Gentner, D. (submitted). MAC/FAC: A structure-mapping model of similarity-based retrieval.

Gentner, D. (1989). The mechanisms of analogical learning. In S. Vosniadou & A. Ortony (eds.) *Similarity and Analogical Reasoning* (pp. 199-241). New York: Cambridge University Press.

Gentner, D. & Forbus, K. (1991). MAC/FAC: A model of similarity-based retrieval. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Gentner, D. Rattermann, M. & Forbus, K. (1993). The roles of similarity in transfer: Separating retrievability from inferential soundness. *Cognitive Psychology*, 25, 524-575.

Gick, M., & Holyoak, K. (1980). Analogical problem solving. *Cognitive Psychology*, 12, 306-355.

Gick, M., & Holyoak, K. (1983). Schema induction and analogical transfer, *Cognitive Psychology*, 15, 1-38.

Holyoak, K. & Thagard, P. (1989) Analogical mapping by constraint satisfaction. *Cognitive Science*, 13(3), 295-355.

Miller, G., Fellbaum, C., Kegl, J. & Miller, K. (1988). WordNet: An electronic lexical reference system based on theories of lexical memory, *Revue Quebecoise Linguistique*, 16, 181-213.

Thagard, P., Holyoak, K., Nelson, G., & Gochfield, D. (1990). Analogical retrieval by constraint satisfaction. *Artificial Intelligence*, 46(3), 259-310.

Wharton, C. Holyoak, K., Downing, P., Lange, T. & Wickens, T. (1991). Retrieval competition in memory for analogs. In *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society* (pp. 528-533). Hillsdale, NJ: Lawrence Erlbaum Associates.

Wharton, C., Holyoak, K., Downing, P., Lange, T., Wickens, T., & Melz, E. (in press). Below the surface: Analogical similarity and retrieval competition in reminding. *Cognitive Psychology*.