

AN EXPLICIT REPRESENTATION OF FORGETTING

Michael T. Cox and Ashwin Ram

College of Computing

Georgia Institute of Technology, Atlanta, GA 30332-0280

email: cox@cc.gatech.edu; ashwin@cc.gatech.edu

Abstract

A pervasive, yet much ignored, factor in the analysis of processing-failures is the problem of misorganized knowledge. If a system's knowledge is not indexed or organized correctly, it may make an error, not because it does not have either the general capability or specific knowledge to solve a problem, but rather because it does not have the knowledge sufficiently organized so that the appropriate knowledge structures are brought to bear on the problem at the appropriate time. In such cases, the system can be said to have "forgotten" the knowledge, if only in this context. This is the problem of forgetting or retrieval failure. This research presents an analysis along with a declarative representation of a number of types of forgetting errors. Such representations can extend the capability of introspective failure-driven learning systems, allowing them to reduce the likelihood of repeating such errors. Examples are presented from the Meta-AQUA program, which learns to improve its performance on a story understanding task through an introspective meta-analysis of its knowledge, its organization of its knowledge, and its reasoning processes.

Keywords

Machine learning; knowledge representation; meta-reasoning; case-based reasoning; blame assignment.

Introduction

Many reasoning tasks can be viewed as memory problems rather than as traditional problem-solving problems. Instead of computing a solution from first principles, a problem may often be solved by remembering a past solution or derivation of a solution, and then adapting the solution to the current problem (Kolodner, to appear). This is a case-based reasoning approach to intelligent behavior. Such methods rely on an effective memory organization which allows the system to retrieve appropriate past solutions to guide the construction of a new solution. The indexing problem (Kolodner, 1984; Schank, 1982; Schank and Osgood, 1990) is the problem of choosing cues, or features in an input, to be used as indexes for retrieving from memory the knowledge structures necessary to process an input. The converse problem, then, is the problem of forgetting. If the cues are not chosen with care during retrieval time, or if the indexes are not chosen well during encoding, then the reasoner will be unable to recall a memory structure when necessary. Thus reasoning failures can occur because of faulty memory organization as well as because of faulty reasoning or faulty knowledge.

If a problem solver or understander treats the indexing problem seriously during planning or comprehension, then it must take the problem of forgetting seriously if it is to learn. A failure-driven learning system gains experience by adjusting its background knowledge (BK) in response to errors, so as to avoid repeating similar failures in the future. As argued in Ram and Cox (1992), the organization of the BK, as well as the BK itself, are possible causes of failures during the reasoning process. Blame assignment involves determining the cause of failure in order to decide what to learn. Thus an analysis of forgetting is essential for effective blame assignment, and for determining how to adjust the organization of memory, so that future retrieval strategies are successful.

The solution for this type of learning is to represent the reasoning process explicitly in structures called Meta-XP's (Ram and Cox, 1992). A Trace Meta-XP (TMXP) is a structure that records a reasoning trace and explains how solutions were generated, whereas an Introspective Meta-XP (IMXP) is a causal pattern that, when applied to a TMXP, explains why these solutions fail. These structures allow direct inspection of the reasoning process and thus facilitate blame assignment. When the reasoning process involves forgetting, however, there is the additional problem of representing the lack of a mental event, such as retrieval failure. This is different from representing actual mental actions; instead, it involves representing what did not occur. To accomplish this, the representation utilizes a modified version of Doyle's (1979) truth values for belief management.

Section 2 of this paper presents a motivational example to illustrate the problem of forgetting. Section 3 provides a specification of the knowledge representation used to capture the functions of memory retrieval. Section 4 discusses the types of forgetting that can be captured by the representation and sketches an outline of how learning algorithms can be mapped from such structures. The paper concludes in section 5 with a discussion of some issues and future research directions.

The Stranded Motorist Example

To illustrate the indexing problem and its relation to the blame assignment problem, consider the following scenario. John, the typical absent-minded type, is going on a long vacation to get away from the city for a weekend. He plans what he needs to do, gets into his car and drives to the store where he will buy food, camping equipment, and fill the car with gas. He enters the store, tries to remember all that he needs to do before going, makes his purchases, and leaves. John then begins the long drive to the mountains. Of course, halfway there he runs out of gas. Luckily he had just purchased a gas can for his portable camping stove, but it is empty. What does John do?

First, he must fix his current situation so that he can continue the trip. The obvious solution is to take the gas can and walk to a station, if he knows that there is one within walking distance. Alternatively, he can hitchhike to a gas station to get gas. He then returns to the car to fill it up. Only then can he continue the vacation. Little information is present in the recovery from this failure that is worth learning, since this solution is easily computed by subgoalting. John may regress the goal solution back to the initial conditions, so that the solution can be cached for future use (c.f., for example, Veloso and Carbonell, 1991). However, since the plan construction is not computationally expensive, the utility of precompiling the solution may outweigh the additional cost of searching for it at failure time. Besides, it provides little benefit to have an efficient solution to the problem, if one can avoid the problem altogether.

In addition to merely recovering from the failure, John is determined not to repeat this mistake again. Though the explanation, "car is out of gas because all gas is used up", is correct, it is not a useful explanation for reducing the likelihood of the recurrence of the failure. So the blame is not contained in the solution to the local failure itself. The global blame is that John forgot to fill up with gas while at the store. In his memory, John had the goal to fill up when he got to the store, however, the goal was not indexed so that the context of being at the store at the start of a long trip would trigger a memory retrieval of the goal. Assuring that this reminding will occur in such circumstances in the future is a memory reorganization problem.

The point of the example is that plan repair depends on not only the background knowledge of the reasoner, but also on the associations that organize the memory of the system. A system may fail, not because it does not know a particular proposition, but because it does not retrieve that knowledge at the proper time. Thus forgetting (i.e., retrieval failure) is a neglected, yet ubiquitous, cause in the blame assignment puzzle.

A Representation of Forgetting

Unfortunately forgetting is not a mental event, but rather the lack of successful memory processing. To be able to represent this state of affairs then, one must have a notation for expressing this non-occurrence. Forgetting can be expressed if the system can represent that it does not believe a successful memory retrieval has occurred. The belief logic of Doyle (1979) has four truth values for a given proposition "p." If p is believed then it is in the set of beliefs, whereas if p is not believed then it is out. Conversely, the negation of the assertion of p may be either in or out of the agent's set of beliefs. Therefore the four values are p(in), p(out), \neg p(in), and \neg p(out). Using these values the system needs to be able to declare that there is a memory item that was not retrieved. Thus it could create a dummy concept representing the forgotten item that it believes did not result from some retrieval process. This concept could

be marked as unbelieved, since it was not retrieved and cannot be specified by the system. But it is incorrect to assert that the concept is not believed if it is in the system's background knowledge (BK). In other words, it is believed but not recalled. However, if we postulate a set of beliefs called the foreground knowledge (FK), representing the working memory of the system, then we can modify the belief logic to claim belief membership with respect to a particular set of beliefs.¹ Thus P, a given memory item that was not retrieved, may be in the set of beliefs with respect to the BK, written $P(\text{in}_{BK})$, but out of the set of beliefs with respect to the FK, written $P(\text{out}_{FK})$.

Forgetting can now be represented as a subset of the class of omission failures described by Cox and Ram (1992). Figure 1 illustrates a graph notation for this simple class of forgetting problems.² The figure shows a core IMXP centered about a Retrieval Failure base IMXP. Numbers on the graph specify the relative temporal order of links that connect nodes in the graph. In this basic representation, forgetting is structurally equivalent to retrieval failure (i.e., retrieval of a memory item was attempted but no item was fetched), though there are numerous possible causes for this effect, leading to multiple types of forgetting.

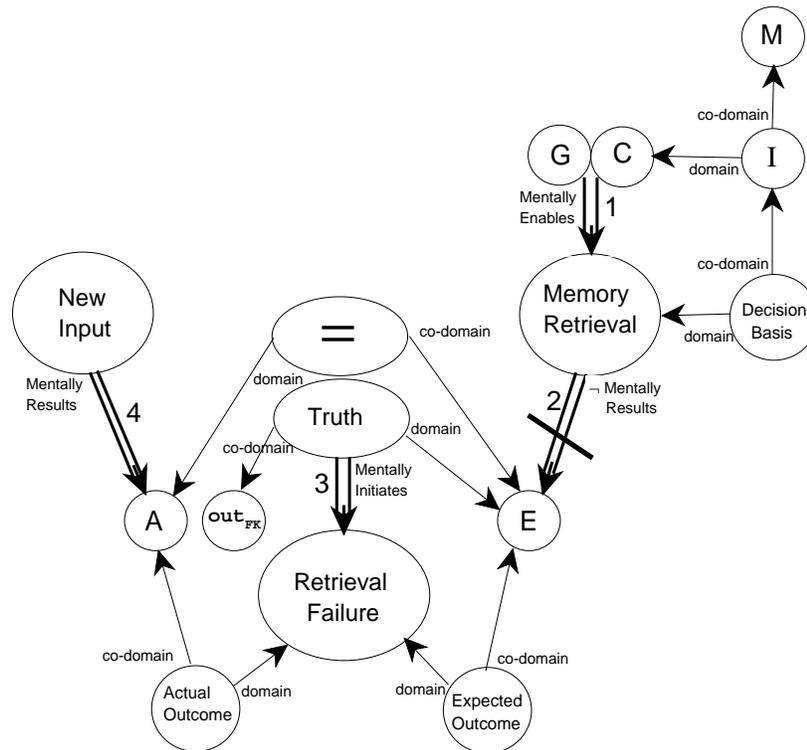


Figure 1. Representation of forgetting

A=actual; E=expected; G=goal; C=context; M=memory item; I=memory index.

The graph representation for forgetting includes a goal (G) to remember (often characterized as a question) plus a context (C) which enables a memory retrieval process. The retrieval is based upon the available indexes (I) that correspond to cues provided by the goal and context. These indexes point to memory items (M). In successful retrieval this process would result in an expectation (E) with a truth value in the set of beliefs with respect to the FK. This memory item is equal to some later actual verified concept (A). In retrieval failure, or forgetting, the memory item M is not retrieved to produce the expectation E. Thus the node E is out of the set of believed concepts with respect to the FK. This condition is designated as $E(\text{out}_{FK})$. Knowing that the node E is out of working memory initiates the state such that the system believes a retrieval failure occurred.

1. Compare this with the assumption maintenance system discussed by McDermott (1989). In general, propositions may be in or out with respect to arbitrary sets of beliefs, which in the Meta-AQUA system, are used to represent what is in the FK during different reasoning experiences.

2. Attribute and relations are represented explicitly. The ACTOR attribute of event X with value Y is equivalent to the relation ACTOR having domain X and co-domain Y.

Types of Forgetting

As specified in Table 1, there can be a number of ways that the memory retrieval process may fail, depending on the conditions of the nodes A, E, G, I, and M. If the memory item is not in the BK (out_{BK}), then there is nothing to be retrieved. This can occur either because there never was a concept in memory to be retrieved, or because the item was previously deleted from memory. Ostensibly there is no difference between the two in this representation. For example, in the Meta-AQUA story understanding system (Cox and Ram, 1992), a novel situation (marked as an absent memory in Table 1) exists when trying to explain a police dog barking at a passenger's luggage in the airport. The system had previously encountered dogs barking only at animate objects, so it had no structure in memory to understand this novel event. This example does not exactly represent forgetting, since there never was a memory item in its experience to be retrieved. In systems that delete memory items in order to facilitate learning (e.g., Markovitch and Scott, 1988), however, trying to remember a deleted item is equivalent to a novel situation.

Table 1: Truth values on graph nodes

Description	A	E	G	I	M
Absent Memory (Novel Situation)	in_{FK}	out_{FK}	in_{FK}	out_{BK}	out_{BK}
Absent Index (Missing Association)	in_{FK}	out_{FK}	in_{FK}	out_{BK}	in_{BK}
Absent Retrieval Goal	in_{FK}	out_{FK}	out_{FK}	\emptyset	\emptyset
Absent Feedback	out_{FK}	out_{FK}	\emptyset	\emptyset	\emptyset

\emptyset = don't care

Forgetting may also occur when an item is in memory but cannot be retrieved, because the indexes provided do not map to the item given the cues of the current goal and current context. This is termed an absent index or missing association (see Table 1). The memory item is not missing; instead, it is lost. There is a tension in blame assignment between ascribing blame to either the context side or the index side. If the environment does not provide sufficient cues, or the reasoner is distracted and so is not focussing on the most favorable portion of the context, then even a good indexing scheme will not retrieve the desired memories. Alternatively, if the indexes chosen to anticipate future use of a knowledge structure are not related to the goals to which the structure might be used, then even in the best context such knowledge will not be retrieved.

Kolodner (to appear) has likened indexed memory retrieval to the indexing of recipes in a cookbook. If one is trying to locate a recipe for a chocolate beverage that originated in the Castilian region of Spain, then it will be useless to attempt to find the recipe using such cues if the recipe is indexed under "hot chocolate" and "dessert." The recipe may be in the book, but it will do no good to look for it under entries for Castilian, Spanish, chocolate, or beverage. Like a forgotten memory, the item will only be found if the indexer has made associations with features likely to be found in the contexts that lead to recipe lookup.

A special case of the above two categories (absent memory and absent index) exists when one examines opportunistic reasoners (e.g., Birnbaum and Collins, 1984; Hammond, 1988; Ram, 1991). Because actions to satisfy goals do not always have their preconditions met at reasoning time, an opportunistic reasoner suspends the goal, indexing the goal in memory so that it can be retrieved at a later time when the conditions become satisfied. In the stranded motorist example, John plans to fill his gas tank before leaving. To perform this action, however, he must be at the location of a gas pump. Thus the goal is suspended until he arrives at the store. When he arrives and tries to remember all of the things he needs to do before starting his journey, the cues in the environment do not correspond to the indexes chosen to map this particular goal in memory. Thus he forgets to perform this action before leaving on his trip. It is only when the car grinds to a halt that John is reminded of his prior plan. Though human conclusions attributing memory problems to an error seem effortless, there must be an examination of the possible causes of the problem. The car could have run out of gas because of a leak in the gas tank. However it is the reminding of the earlier reasoning (in Meta-AQUA this is stored in a TMXP) which elicits the realization that the blame lies in the fact that the goal was not retrieved at the gas station. An intelligent system then must be able to realize that the association between starting long trips and filling up with gas is an important connection to retain. Thus in future instances of this situation (going

on long trips) it will be reminded of the current failure and check the gauge.

Alternatively, John may never have planned to fill up the car to begin with, thus never generating the goal to perform such action. The situation is similar to the absent memory (novel situation) condition. There is no goal in memory that can be retrieved to remember what to do. Instead of a memory failure, the blame lies with the planner instead. In like manner, John may have generated the goal to fill the car and indexed it in memory to be retrieved later, but then he never tried to remember what he had planned earlier. This situation corresponds to the absent retrieval goal row of Table 1. This problem is also with the reasoner, since it never generated the goal to perform memory retrieval.

The final category of forgetting is impossible to reason about a priori. This case has no feedback to remind the reasoner that there was an opportunity to retrieve an appropriate memory. These are lost opportunities which might have presented the agent with additional knowledge. Because the node A is missing, however, there can be no hope of knowing that this opportunity existed. There must be either an inferred or input concept to remind the reasoner through hindsight that a memory failure occurred. Returning to John, if he forgets to fill up but does not run out of gas before the next gas station, then there is likely no reminding, and hence no opportunity to learn.

The benefit of representing a reasoning trace with Meta-XP's is that the system can reflect on the structure of its own reasoning. Furthermore, it can pose questions on such reasoning. For example, using this approach to forgetting the system can pose questions on a node such as I, asking itself in response to retrieval failure: Is the index wrong? If the index is flawed, the system can hypothesize that there really exists a memory, $M \in BK$. A question to this effect is posed on node M, the question is indexed, the reasoning that led to this question stored in a TMAP is tagged off the question, and the process is suspended. This allows opportunistic reasoning at the introspective level. If and when the system is reminded of the question in the future, the question can be resumed and checked to see if the reminding confirms the existence of this structure. Memory reindexing can then be performed in light of the reasoning that gave rise to the failure.

Discussion

The analysis above can be expanded to more complicated situations, but space limitation does not permit adequate review here. Briefly though, given a reconstructive memory (Schank, 1982) such that memory items are not retrieved as whole objects but must be reconstructed from partial memories, there may be partial matches. For example, John may have remembered that he had planned to do something (i.e., he had a goal), but could not remember what the goal was. Thus the memory system may retrieve a goal without a goal object specification. Or John may remember that he needs to do something before he leaves, and that it is an action related to the car, but the specific act may be forgotten. Thus there is an issue concerning the quality of match between that which is retrieved and that which is the actual target. In particular, it is uncertain at what point a system can assert that an item was retrieved or not retrieved in reconstructive memories.

Furthermore, the analysis of the preceding sections does not address the possibility that the retrieval of one memory item (for example a solution to a problem) may block the retrieval of other structures. A liberal reasoner may have a bias toward the first solution retrieved, whereas a more conservative or cautious reasoner may search for additional items. It is an open question whether a bias toward the first memory that "comes to mind" should be viewed as a kind of forgetting when further memories are more appropriate given the context.

Some related research has been performed in the Artificial Intelligence community.³ Schank (1982) has discussed forgetting in general terms he refers to as mashing⁴, which is a type of similarity-based learning. As an intelligent agent processes multiple occurrences of similar concepts, the similar items are collected and details of individual experiences are lost. However no attention to the implications of these lost items is given. A number of systems use deletion or decay to remove memories with low usefulness (e.g., Markovitch and Scott, 1988; McClelland and Rumelhart, 1986). Other systems such as Soar (Tambe et al, 1988) analyze the expected utility of a learned concept to determine whether or not to store it at all. However, none of these systems present a theory of what to do when a memory is missing or cannot be found, nor do the systems have any potential for learning from such situations. No previous system has emphasized the effects of forgetting, explained why one must directly reason about it, or

3. An interesting related research topic in the psychological community is that on metamemory and the tip of the tongue phenomena whereby $(p \in BK) \in FK$ yet $p \notin FK$, i.e., one knows that one knows a fact yet cannot recall it. For an overview see Schneider and Pressley (1989).

4. Carbonell (1983) briefly discusses a similar effect he refers to as fading.

presented an explicit syntax for representing it.

The thrust of this paper is the knowledge representation of forgetting and the arguments of why representation is important, rather than being a detailed explanation of how these structures are reasoned about or the details of learning which result. The claim is that a reasoner must be able to not only reason about what happens and why, but also what did not happen and why not. An analysis of such can point to what should have happened. A comparison of what should have happened and what did happen can then aid blame assignment. Blame assignment determines the cause of failure, and thus points to changes in the reasoner's knowledge and the organization of its knowledge, so as to avoid repeating such mistakes in similar future situations. An intelligent agent learns from mistakes; a fool is doomed to repeat them indefinitely.

Acknowledgments

This research was supported by the National Science Foundation under grant IRI-9009710 and by the Georgia Institute of Technology. The authors wish to thank Janet Kolodner for criticism and discussion, and Sue Ferrell for proofing a draft of this paper. Helpful discussion and comments were also offered by the IGOR Research Group.

References

- Birnbaum, L. and Collins, G. (1984); Opportunistic Planning and Freudian Slips; *Proceedings of the Sixth Annual Conference of the Cognitive Science Society*; University of Colorado, Boulder, CO (pp. 124-127)
- Carbonell, J. G. (1983); Learning by Analogy: Formulating and Generalizing Plans from Past Experience; *Machine Learning: An Artificial Intelligence Approach*, Vol. 1 (eds. R. Michalski, J. Carbonell and T. Mitchell); Morgan Kaufmann Publishers, Inc., Los Altos, CA.
- Cox, M. T., and Ram, A. (1992); Multistrategy Learning with Introspective Meta-Explanations; *Machine Learning: Proceedings of the Ninth International Conference (ML92)*, (eds. D. Sleeman and P. Edwards); Morgan Kaufmann, Los Altos, CA.
- Doyle, J. (1979); A Truth Maintenance System; *Artificial Intelligence*, Vol. 12, (pp. 231-272)
- Hammond, K. (1988); Opportunistic Memory: Storing and Recalling Suspended Goals; *Proceedings of the Workshop on Case-Based Reasoning (DARPA)*; Morgan Kaufmann Publishers, Inc., San Mateo, CA.
- Kolodner, J. L. (1984); *Retrieval and Organizational Strategies in Conceptual Memory: A computer model*; Lawrence Erlbaum Associates, Publishers, Hillsdale, NJ.
- Kolodner, J. L. (to appear); *Case-Based Reasoning*; Morgan Kaufmann Publishers.
- McDermott, D. (1989); A General Framework for Reason Maintenance; Technical Report 691, Yale University, Department of Computer Science.
- Markovitch, S. and Scott, P. D. (1988); The Role of Forgetting in Learning; *Proceedings of the Fifth International Conference on Machine Learning*, Ann Arbor, MI.
- McClelland, J. L. and Rumelhart, D. E. (1986); A Distributed Model of Human Learning and Memory; *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 2, (eds. J. L. McClelland and D. E. Rumelhart), (pp. 170- 215)
- Ram, A. (1991); A Theory of Questions and Question Asking; *Journal of the Learning Sciences*, Vol. 1, No. 3 & 4.
- Ram, A., and Cox, M. T. (1992); Introspective Reasoning Using Meta-Explanations for Multistrategy Learning. *Machine Learning: A Multistrategy Approach IV*, (eds. R. S. Michalski. and Tecuci, G.); Morgan Kaufmann.
- Schank, R. C. (1982); *Dynamic Memory: A theory of reminding and learning in computers and people*; Cambridge University Press, Cambridge, MA.
- Schank, R. C., and Osgood, R. (1990); A Content Theory of Memory Indexing; Technical Report 2. Institute for the Learning Sciences, Northwestern University, Evanston, IL.
- Schneider, W. and Pressley, M (1989); *Memory Development Between Two and Twenty*; Springer-Verlag.
- Tambe, M, Newell, A. and Rosenbloom, P. S. (1990); The Problem of Expensive Chunks and Its Solution by Restricting Expressiveness; *Machine Learning*, Vol. 5, (pp. 299-348)
- Veloso and Carbonell, J. G. (1991); Automating Case Generation, Storage and Retrieval in PRODIGY; *Proceedings of the 1st International Workshop on Multi-Strategy Learning*, (eds. R. S. Michalski, and G. Tecuci, G.), (pp.363-377)