

A METAMEMORY MODEL FOR AN INTELLIGENT TUTORING SYSTEM

UN MODELO DE METAMEMORIA PARA UN SISTEMA TUTOR INTELIGENTE

Manuel F. Caro¹, Darsana P. Josyula²

Recibido para publicación: 22 de mayo de 2014 - Aceptado para publicación: 5 julio de 2014

ABSTRACT

Metamemory refers to the processes involved in self-regulation or self-awareness of memory. In this paper we describe a novel rule-based architecture of metamemory named M²-Acch. M²-Acch consists of a cycle of reasoning about events that occur in long-term memory (LTM) in an intelligent tutoring system. M²-Acch is composed of a three layer structure: static layer, functional layer and information layer. The structural components of each layer model are described using formal definitions. M²-Acch uses confidence judgments for recommending search strategies for adaptation to changes in the information retrieval constraints. An intelligent tutoring system named FUNPRO was implemented and validated. The results of the experimental tests show that M²-Acch can be used as a valid tool for adapting to changes in the constraints in information retrieval from long-term memory in artificial cognitive systems.

Keywords: metamemory; intelligent tutoring system; self-monitoring; metacognition

RESUMEN

Metamemoria se refiere a los procesos que intervienen en la autorregulación o la auto-conciencia de la memoria. En este trabajo se describe una nueva arquitectura basada en reglas de metamemoria llamado M²-Acch. M²-Acch consiste en un ciclo de razonamiento acerca de los eventos que ocurren en la memoria a largo plazo (MLP) en un sistema tutorial inteligente. M²-Acch se compone de una estructura de tres capas: la capa estática, capa funcional y la capa de información. Los componentes estructurales de cada modelo de capas se describen utilizando definiciones formales. M²-Acch utiliza juicios de confianza para recomendar estrategias de búsqueda para la adaptación a los cambios en las limitaciones de recuperación de información. Un sistema tutorial inteligente llamado FUNPRO fue implementado y validado. Los resultados de las pruebas experimentales demuestran que M²-Acch puede ser utilizado como una herramienta válida para la adaptación a los cambios en las restricciones en la recuperación de información de la memoria a largo plazo en los sistemas cognitivos artificiales.

Palabras Clave: metamemoria; Sistema tutor inteligente; auto monitoreo; metacognicion

¹Departamento de Informática Educativa, Universidad de Córdoba. Montería, Colombia, mfcarop@unal.edu.co

²Department of Computer Science, Bowie State University, Bowie, MD – USA, darsana@css.umd.edu

1. INTRODUCTION

Retrieval is one of the basic processes of memory activities [1]–[3] and has been studied for many years e.g., [4]. Retrieval includes tasks associated with accessing of stored information. One crucial influence on the outcome of any retrieval process is the knowledge available to that process [5]. This knowledge includes search constraints [6], parameters and other information related to the target of the search [7]. The search constraints restrict the information retrieved by the search task, by influencing the search strategy used to fulfill the search task goal [8].

Changes in search constraints affect the performance of information retrieval [6], [9]. When there are changes in constraints, information retrieval cannot be done effectively by the same search strategy for all cases. Thus the system needs to assess changes in the constraints of the search tasks and select the most appropriate search strategy.

In the context described above, we propose the use of metamemory as a mechanism that allows adaptation to changes in the constraints of a search task. Metamemory is a component of metacognition [10]. As such, it deals with the memory capabilities of a system and strategies that aid Self-adaptive memory [11], [12], as well as the processes involved in self-monitoring of the memory. In our approach, adaptation is retrieved by first monitoring and identifying changes in the constraints of the search task; once a change in the constraints is detected, then the metamemory assesses the knowledge available to satisfy the constraint and suggests the most appropriate search strategy. In order to monitor and identify changes in the constraints, the meta-level stores expectations about the object-level information like judgments, observations and memory events.

The paper is structured as follows. In Section II, we describe the proposed metamemory model named M2-Acch. Section III presents the

implementation and evaluation of M2-Acch in an Intelligent Tutoring System (ITS). Finally, we present the conclusions of the work.

2. Metamemory model for adapting to constraint changes (M2-Acch)

M2-Acch (pronounced “match”) is a Metamemory Model and consists of a cycle of reasoning about events that occur in long-term memory (LTM) in intelligent tutoring system, see Figure 1. The reasoning cycle inputs are the memory events that occur in LTM and the output consists of recommendations which may vary according to the memory events. In particular we will focus on the reasoning process that allows adaptation to constraint changes related to retrieving information from LTM.

M2-Acch comprises a two-layer modeling. The static layer contains the structural model that describes the static relationships between the elements of the metacognitive architecture. The dynamic layer comprises the functional model that describes the processes of metamemory and decision making

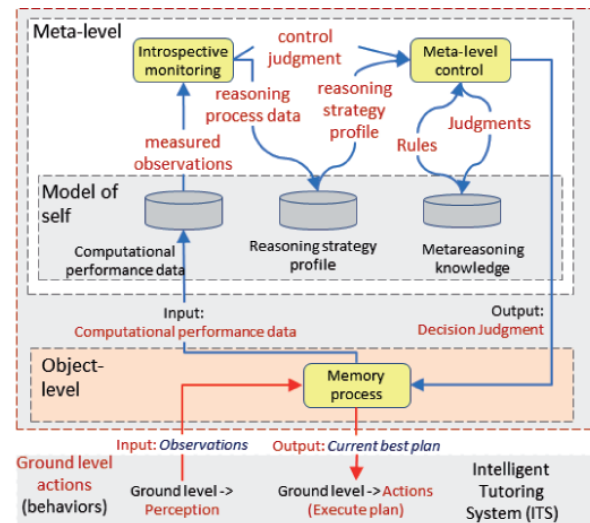


Figure 1. Cycle of reasoning in M2-Acch

The static layer

In this section, we introduce the basic concepts of the formalism to describe the structural model of M2-Acch.

Definition 1. $T=\{L, E, G, D, J, S\}$ is the set containing the basic types of elements of the meta-memory model, where:

L represents the structural levels of the model.

E describes the events that occur in a system that directly affect LTM.

G is the goal of an event.

D represents the constraints required to perform a search.

J represents the judgments that the meta-level triggers about events E that happens at the object-level.

S is the search strategy recommended by the meta-level.

Definition 2. M²-Acch is composed of two main levels (L): *object-level* and *meta-level*. The object-level contains the LTM of the cognitive system and the meta-level contains *knowledge about LTM* and *processes for reasoning about LTM*.

Definition 3. The events (E) represent actions that are performed on the memory. $E=\{ID, y, g, d, t\}$ is the set of components that represents the structure of an event, where:

ID is the unique identifier of the event.

y is the type of the event, $y \in Y$ and

$Y=\{\text{call, execute, re-configure}\}$.

g is the goal of the event.

d is the constraint of the event.

t is the memory task that originated the event.

Definition 4. Goals (G) are subcomponents of events. Each event can have only one goal. Goals contain relevant meta-data about information to be stored or retrieved from memory.

Definition 5. The constraints (D), in an event (E) refer to the information requirements that must be

satisfied so that the *event fulfills* the goals.

Definition 6. Metacognitive judgments (J) represent assessments performed in the meta-level about events that occur in memory. These judgments provide information that the system uses to determine whether it is able to attempt retrieval or storage. In the current model, we have two types of metacognitive judgments, these are:

COP (Certainty of Optimal Performance) measures the degree of certainty that the system has with regarding to optimum performances obtained in the past, having constraints similar to the current user.

CSRD (Certainty of Satisfying the Retrieval constraints) measures the degree of certainty that the system has with regard to the level of knowledge that the system possesses to attend the requirements of the retrieval constraints.

Definition 7. The meta-level contains a schema with information search strategies (S) available at the object-level. A major meta-level function is to recommend the most appropriate search strategy for the constraints of information retrieval from LTM.

Definition 8. The expectations are behaviors that the cognitive system is expecting to pass in object-level. Expectations are associated to sensors. The sensors are elements that the meta-level has to monitor information flows coming from the object-level. The information contained in the sensor is called observation. When observations do not match with expectations, then there is a violation of expectation. The violations of expectations are interpreted by the meta-level as failures in the reasoning of object-level.

Dynamic layer

This section describes the processes for monitor and control of LTM that have been incorporated into M2-Acch. Monitoring processes include mechanisms for detecting events in LTM and performing deep search processes on the meta-level knowledge about the object-level.

Moreover, control processes include rules for the recommendation of search strategies on LTM.

3. Introspective Monitoring of memory

The proposed model has the following three mechanisms to monitor LTM: *memory event detection*, *metamemory judgment* and *in-depth search*.

Memory event detection: In a cognitive system, when a process calls a search task in the memory, then a memory event is triggered. The meta-level stores traces of all the events that occur in LTM.

Rule 1. When some task calls a search task in LTM, then a memory event is triggered.

Rule 2. All constraints of a search task are associated with the corresponding memory event that is triggered.

Rule 3. If the information constraints of an event are different from the constraints required to execute a search by default, then the meta-level detects a change in the constraints of the event.

Reasoning in-depth and Metamemory judgment:

Rule 4. If any change in the constraints of an information retrieval task is detected in event memory, then the meta-level decides to launch a deeper reasoning process. The reasoning involves the examination and assessment of the performance of the information retrieval task with similar constraints in the past.

Rule 5. If at least one strategy that produces an optimal performance is found in the history of events, then a COP judgment with value – high is triggered.

Rule 6. If it is not found at least one strategy in the history of events that produces an optimal performance, then it is triggered a COP judgment with value - low.

Rule 7. If a COP judgment is triggered with a low value, then the meta-level assesses the degree of knowledge that possesses to satisfy the constraint.

Rule 8. If some kind of knowledge to satisfy the task constraints is retrieved, then the meta-level triggers a CSRD judgment with value - high.

Rule 9. If no knowledge to satisfy the task constraints is retrieved, then the meta-level triggers a CSRD judgment with value - low.

Rule 10. If observations in sensors do not match with expectations, then there is a violation of expectation. The violations of expectations are interpreted by the meta-level as failures in the reasoning of object-level.

Metalevel Control of memory

Rule 11. If a COP judgment has high value then the strategy with optimal performance in the event history is recommended.

Rule 12. If a CSRD judgment has high value then a strategy with support for knowledge in constraints is recommended.

Rule 13. If the CSRD judgment has low value then it is recommended to stop the memory search task.

4. IMPLEMENTATION AND VALIDATION

In this section, we describe the features of FUNPRO, the Intelligent Tutoring System designed to assess the M²-Acch metamemory model and how constraint changes are implemented in FUNPRO for information retrieval. Then, the performance metric used for measurement are depicted, followed by a description of overall validation procedure.

FUNPRO

We have designed an ITS named FUNPRO (FUNdamentos de PROgramación) using MODESEC methodology [13]. FUNPRO is an Intelligent Tutoring System for teaching *Introduction to Programming in Engineering*, Figure 2.

Identification of call to retrieve tasks with constraint changes in FUNPRO

FUNPRO has implemented three types of search strategies: (1) matching simple query, the search



Figure 2. Cycle of reasoning in M2-Acch

query in a simple SQL type; (2) exclusive search is similar to (1), but excludes some results and; (3) vote-based search, this strategy is based on the nearest neighbor algorithm.

FUNPRO has a function named `playResource` that is responsible for retrieving the URL of learning resources from the knowledge base, and deploying them in the lesson. However, the restrictions that FUNPRO generates to search for resources for the lesson are dynamic according to several criteria described below.

Case A. The student enters the lesson for the first time; therefore FUNPRO has only collected information about the student's learning style to recommends learning resources and teaching strategies for the lesson. Thus `playResource` function receives a single constraint.

Case B. FUNPRO finds a resource that meets the restrictions of the search, but for some reason cannot be deployed in the lesson, for example: the resource URL is broken. In this case FUNPRO has to retrieve from the knowledge base another resource that supports the student's learning style, but it has to exclude the resource with the URL broken.

Case C. The student evaluates the resource after use. If the resource has received a poor evaluation, the system recommends new resources that have been well evaluated by students with similar characteristics to the current.

Case D. Case D has two variants. In the first, if the student obtains a low performance in the lesson, then the system remains in the current lesson but reconfigures strategies for teaching and learning resources. In the second variant, if the student obtains a high performance; then the system presents a new lesson.

Metamemory in FUNPRO

5. INTROSPECTIVE MONITORING

When a new memory event trace is stored in the meta-level, the monitoring process starts as the meta-level detects and identifies the event in LTM.

If the event detected is a call to a search task (*e. g. a new search for learning resource recommendation*), then the meta-level checks for changes in task constraints of the target of the search. The changes in the task constraints occur when there is a failure due to difference between the observation and expectation of target of search. Expectations in FUNPRO can be specified by default in the system configuration or may be self-generated by the system task (*e. g. when a new learning resource is recommended then the system aspects that the resource will be useful for student learning*).

If any change in constraints of the search task is detected (*rule 4*), then the meta-level decides to launch a deeper reasoning process about the memory event. The reasoning involves examination and assessment of the performance of the information retrieval task with similar constraints in the past. In the examination and assessment of the performance process the meta-level searches for events that occurred in the past with similar restrictions (*rules 5-10*).

If the events with similar meta-level constraints are located, meta-level then proceeds to obtain the search strategies that have been used to process such events. If the meta-level has at least one event that has been processed successfully, then it makes a COP judgment with high value (rules 6-7). This means that the meta-level in FUNPRO has a high level of certainty of knowing the appropriate search strategy to satisfy the request of retrieving information contained in the current event. To that purpose the meta-level maintains a performance profile of search tasks, which consists of a record of the search strategies that have been used to process information retrieval requests in the past. In this performance profile, information that shows whether the search strategy could retrieve the required information is also stored.

6. METALEVEL CONTROL OF MEMORY

The meta-level control is based on the value of the metacognitive judgments. For example, if a COP judgment has a high value, then the meta-level recommends the search strategy that has had better performance in events with similar constraints in the past.

In case the judgment has a low value, and the system has available intelligent complex search strategies, then M2-Acch offers the possibility for the meta-level to recommend these strategies. For this purpose, the meta-level evaluates the knowledge about the requirements implicit in the constraints of the search. If some knowledge related to constraint is obtained, then the meta-level triggers a CSRD judgment with high value. Otherwise, the meta-level control mechanism recommends to the object-level to stop the information retrieval, because there is not enough knowledge to process the search.

Performance metrics used for validation

The validation of the implementation of meta-memory model in FUNPRO on cases A and B are presented next. Since the primary purpose of

the M²-Acch is to monitor and control failures in the reasoning process about information retrieval in cognitive systems, we use the reasoning failures dimension as performance metric of the metacognitive capacity of the system. The metric represents retrieval performance [14] on the number of available resources that were recommended for the lesson. A resource available is one that can be deployed in a lesson, Table I provides a description of the metric for performance evaluation of ITS with metamemory functions.

Table 1. Performance metrics

Metric	Description
ART	% of available resources in retrieval
URT	% of unavailable resources in retrieval

Process

For validation, we generated 50 student profiles with random assignment of learning styles. Then 400 educational resources profiles were generated, 20 educational resources for each one of the 20 pedagogical tactics supported by FUNPRO. For each student profile, a recommendation of learning resources required for the lesson, based on the learning style was generated.

The simulation of the recommendation process was conducted in eight sessions. In each session the number of unavailable resources in the resource base was gradually increased, see Table II for details. Finally, each session was repeated five times to observe the behavior of the meta-level.

Data analysis and discussion

Figure 3 in Section A shows the results of the comparison between the performance of FUNPRO not using metamemory and using metamemory.

It is noted that when including metamemory, FUNPRO shows a low sensitivity to the progressive increase of unavailable resources in the resource base. This means that FUNPRO

Table 2. Session configuration

Session	# of Students	# of resources	# available	# unavailable
1	50	400	360	40
2	50	400	320	80
3	50	400	280	120
4	50	400	240	160
5	50	400	200	200
6	50	400	160	240
7	50	400	120	280
8	50	400	80	320

can adapt to such situations because it is able to select the appropriate search strategy in the event of failures in the information retrieval. Thus when performing information retrieval based on the student's learning styles and needed learning resources, then FUNPRO excludes the resource that is not available for future searches. FUNPRO presents the excluded resources to the system manager (the teacher) in order to review the cause of the problem. If the problem is solved by the system manager then the resource becomes available for future searches.

The results obtained in the experimental tests show that M²-Acch is able to make adaptations in the search strategies. Adaptations are based on changes in the constraints of information retrieval and allow the system to recognize and prevent failures in the recommendations. Therefore metamemory increases the robustness in terms of failure tolerance in information retrieval from LTM.

CONCLUSION

This paper has presented a novel architecture that involves the use of metamemory for adapting to changes in the constraints of information retrieval tasks from long-term memory. M2-Acch provides mechanisms for monitoring, identifying and making decisions about the events related to information retrieval that occur in long-term memory. The novel architecture is able to identify changes in constraints affecting information retrieval from long-term memory. M2-Acch has a system based on confidence judgments for recommending search strategies that allow adaptation to changes in the information retrieval constraints.

M2-Acch was implemented in Prolog and validated in an intelligent tutoring system called FUNPRO. The results of the experimental tests show that FUNPRO was able to adapt to changes in the restrictions on the search effort. FUNPRO performance using M2-Acch was superior in terms of retrievals available in comparison to the performance of FUNPRO without M2-Acch. The results of the tests show that M2-Acch is a valid tool for improving the process of information retrieval from long-term memory in cognitive systems.

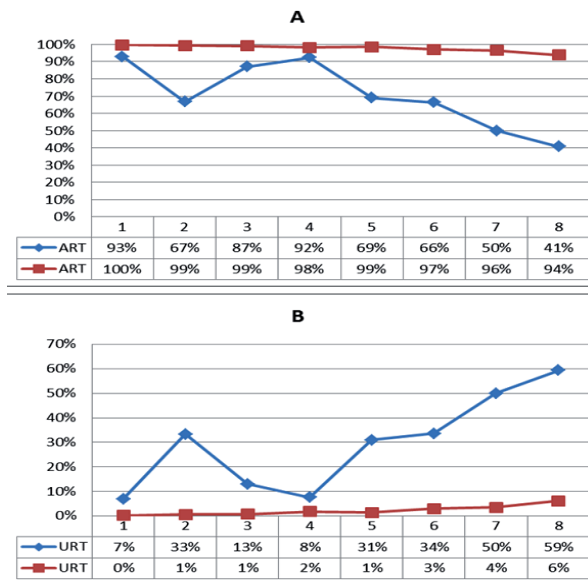


Figure 3. Comparison between retrieval rates in FUNPRO: Blue line: retrieval rate without metamemory; Red line: retrieval rate with metamemory. Section A shows the comparison between available retrieval rates. Section B shows the comparison between unavailable retrieval rates. Horizontal axis represents sessions described in Table II.

REFERENCES

- [1] R. Schank, *Dynamic memory: A theory of reminding and learning in computers and people*. Cambridge, MA, 1982.
- [2] E. Düzel, R. Cabeza, T. W. Picton, a P. Yonelinas, H. Scheich, H. J. Heinze, and E. Tulving, "Task-related and item-related brain processes of memory retrieval.," in *Proceedings of the National Academy of Sciences of the United States of America*, 1999, vol. 96, no. 4, pp. 1794–9.
- [3] J. Metcalfe and J. Dunlosky, "Metamemory," *H. L. Roediger III (ed.), Cogn. Psychol. Mem. Vol. [2] Learn. Mem. A Compr. Ref.*, vol. 4 vols. (J, 2008.
- [4] H. Ebbinghaus, *Memory: A contribution to experimental psychology*. 1962.
- [5] D. B. Leake, "Representing Self-knowledge for Introspection about Memory Search A Planful Framework for Internal Reasoning," in *AAAI Spring Symposium on Representing Mental States and Mechanisms*, 1995.
- [6] J. Kizilirmak, F. Rösler, and P. Khader, "Control processes during selective long-term memory retrieval.," *Neuroimage*, vol. 59, no. 2, pp. 1830–41, Jan. 2012.
- [7] N. Unsworth, "On the division of working memory and long-term memory and their relation to intelligence: A latent variable approach.," *Acta Psychol. (Amst.)*, vol. 134, no. 1, pp. 16–28, May 2010.
- [8] A. Mecklinger, "The control of long-term memory: brain systems and cognitive processes.," *Neurosci. Biobehav. Rev.*, vol. 34, no. 7, pp. 1055–65, Jun. 2010.
- [9] N. Huet and C. Mariné, "Memory strategies and metamemory knowledge under memory demands change in waiters learners," *Eur. J. Psychol. Educ.*, vol. XII, no. 1, pp. 23–35, 1997.
- [10] M. Cox, T. Oates, and D. Perlis, "Toward an Integrated Metacognitive Architecture," in *Advances in Cognitive Systems: Papers from the 2011 AAAI Fall Symposium (FS-11-01)*, 2011, pp. 74–81.
- [11] J. A. Crowder and S. Friess, "Metacognition and Metamemory Concepts for AI Systems," in *International Conference on Artificial Intelligence*, ICAI 2010, 2010.
- [12] J. Flavell and H. Wellman, "Metamemory," in *Perspectives on the Development of Memory and Cognition (pp. 3-33)*. Hillsdale, NJ: & J. W. H. (Eds. . R. V. Kail, Jr., Ed. Lawrence Erlbaum Associates, 1977.
- [13] M. F. Caro, R. E. Toscano, F. Hernández, and M. E. David, "Diseño de software educativo basado en competencias," *Cienc. e Ing. Neogranadina*, vol. 19, no. 1, pp. 71–98, 2009.
- [14] S. Ghetti, K. E. Lyons, F. Lazzarin, and C. Cornoldi, "The development of metamemory monitoring during retrieval: the case of memory strength and memory absence.," *J. Exp. Child Psychol.*, vol. 99, no. 3, pp. 157–81, Mar. 2008.