Active Case-Based Reasoning for Lessons Delivery Systems

Rosina Weber[1,2], David W. Aha[2], L. Karl Branting[1], J. Robert Lucas[3], and Irma-Becerra Fernandez[4]

[1]Department of Computer Science, University of Wyoming, Laramie, WY 82071
[2]Navy Center for Applied Research in AI, Naval Research Laboratory, Washington, DC 20375
[3]Joint Center for Lessons Learned, USACOM Joint Warfighting Center, Suffolk, VA 23435
[4]Florida International University, Decision Sciences and Information Systems, Maimi, FL 33199
{weber,aha}@aic.nrl.navy.mil, karl@uwyo.edu, lucasr@jwfc.acom.mil, becferi@fiu.edu

Abstract

Exploiting lessons learned is a key knowledge management (KM) task. Currently, most lessons learned systems are *passive*, stand-alone systems. In contrast, practical KM solutions should be *active*, interjecting relevant information *during* decision-making. We introduce an architecture for active lessons delivery systems, an instantiation of it that serves as a monitor, and illustrate it in the context of the conversational case-based plan authoring system HICAP (Muñoz-Avila et al., 1999). When users interact with HICAP, updating its domain objects, this monitor accesses a repository of lessons learned and alerts the user to the ramifications of the most relevant past experiences. We demonstrate this in the context of planning noncombatant evacuation operations.

1. Introduction

A central focus of knowledge management (KM) (Decker and Maurer, 1999) is the development of *learning organizations* (Senge, 1990), in which each employee's individual knowledge becomes organizational knowledge that is made available in a *knowledge repository*. These repositories can assist in crucial organizational KM tasks, including identifying, eliciting, creating, organizing, classifying, representing, updating, sharing, reusing, and adapting organizational knowledge. Lessons learned (LL) from working experiences are a particularly important form of organizational knowledge. By recording both successes and failures, LL repositories can make past experiences available to improve organizational processes.

LL repositories are maintained by many commercial organizations and are omnipresent in military organizations. Officers involved with decision-making are typically required to write detailed reports on their actions in operations, including descriptions of context, observations, lessons, and recommendations. In the USA alone, over a dozen military LL repositories can easily be found on the Internet, and many more exist on classified networks. For example, repositories are maintained by the Joint Center for Lessons Learned, the Navy Center for Lessons Learned, and the Center for Army Lessons Learned.[1] These centers employ several contractors and civilians to assist with the processes of advertising, lesson collecting, database management/maintenance, and software support.

In contrast to the extensive efforts that have been directed to building LL collections, comparatively few efforts have addressed how to communicate the experiences embodied in these collections to decision makers. Most LL repositories are passive standalone systems and assume that decision makers requiring LL knowledge (1) have the time to search for appropriate lessons, (2) know where to find them, (3) know (or have time to learn) how to use the repository's retrieval software, and (4) can correctly interpret lessons once they have been retrieved. These assumptions are generally unrealistic; decision-makers typically do not have the time, motivation, or skills necessary to exploit LL repositories. Our interviews with many active and retired officers, civilian employees, and contractors suggest that most military LL repositories are viewed by potential users as useless. In our view, a useful lessons learned delivery process must be active: the repository should alert the decision maker as needed in the context of the decision-making process. We refer to this as an *active lessons delivery* process.

We introduce a general architecture for this approach and illustrate its use in a military planning process in the context of HICAP (Hierarchical Interactive Case-based Architecture for Planning) (Muñoz-Avila et al., 1999), a conversational case-based reasoning (CCBR) system, applied to noncombatant evacuation operations (NEOs) (See Section 4.2). We will detail how a LL module can monitor an incremental planning process, compute the relevance of stored lessons, and alert the decision maker with relevant lessons and recommendations.

[1] See www.aic.nrl.navy.mil/~aha/lessons for a list of on-line lessons learned repositories.

2. JULLS: The Joint Universal Lesson Learned System

Here we focus on illustrating the active application of lessons from the Joint Universal Lesson Learned System (JULLS),[1] which is maintained by the Joint Warfighting Center's Joint Center for Lessons Learned (JCLL). The JCLL manages all aspects of JULLS (e.g., collection, analysis, representation, retrieval system, maintenance), which is intended to support military decision makers in the USA Joint Forces Command. It currently contains 908 lessons. We obtained a subset of 150 unclassified lessons that were previously provided to the Armed Forces Staff College. Among these, 33 lessons concern NEOs.

All lessons in JULLS are collected using the Joint After-Action Reporting System (JAARS), each represented by 43 attributes that were selected to facilitate lesson representation, retrieval, and management. Example attributes include the lesson's title, source, classification, sponsor, and the dates on which it was recorded and last edited. The most useful attributes for retrieving lessons are the following text fields:

- *Keywords:* A set of keyword phrases.
- *UJTL Task:* The task referred to by this lesson in the Universal Joint Task List (UJTL).
- *Observation*: A concise text summary of the context and lesson.
- *Discussion*: A multi-paragraph text field describing the lesson's context.
- *Lesson learned:* A concise text summary of the lesson.
- *Recommended action:* Brief text defining how to interpret this lesson in future contexts.

Users interact with JULLS via a commercial search engine. Queries are logical combinations of text inputs (i.e., for 28 of the 43 attributes), and the user can also browse the entire repository. Although helpful, keyword indexing can be problematic. (See Rose and Belew (1991) for a discussion of the limitations of information retrieval techniques.) For example, the commercial search engine currently being used for JULLS does not perform stemming, synonym analysis, or support forms of semantic retrieval. Thus, the user must carefully formulate queries. Furthermore, even if JULLS was connected with the UJTL, its text representations for lessons do not relate, in a computational form, the lesson's target task and the lesson itself. This limits the utility of simple query retrieval search procedures.

In summary, JULLS has the following typical characteristics for LL repositories:

- *Standalone*: JULLS is a standalone system; it is not embedded in a decision support tool.
- *Passive*: JULLS requires decision makers to search it; it cannot proactively provide lessons.
- *Text representation:* The important attributes of its lessons are in text format.

We address these potential limitations in the following sections.

[1] Access to the JCLL WWW site can be requested at www.jtasc.acom.mil.

3. Knowledge Management and Case-Based Lessons Delivery Systems

There is a growing interest among members of the artificial intelligence community in KM. For example, the 1997 Knowledge-Based Systems for Knowledge Management Enterprises Workshop and a recent International Journal of Human-Computer Studies (Decker and Maurer, 1999) special issue both focused on AI and KM.

A promising AI technology for KM is case-based reasoning (CBR), which concerns the structured storage, retrieval, reuse, revision, and retention of information (Watson, 1997). CBR has been applied to several commercial KM tasks, and in particular has contributed to improvements in customer service (Davenport and Prusak, 1998) and text retrieval (Weber, 1999). The growing interest of KM among CBR practitioners and researchers inspired several workshops during the summer of 1999, including the AAAI Workshop *Exploring Synergies of Knowledge Management and Case-Based Reasoning* (Aha et al., 1999).

Although CBR seems a good match, it has rarely been used for lessons delivery systems. Becerra-Fernandez and Aha (1998) propose the use of CBR for a related KM task, but perhaps the only mature application is NASA-Goddard's RECALL (Reusable Experience with Case-Based Reasoning for Automating Lessons Learned), also a standalone system (Sary and Mackey, 1995). A key reason for the paucity of case-based LL systems is that LLs are generally expressed as free text, whereas CBR systems generally require a featural or relational case representation.

Textual CBR is a recent research area within CBR that addresses the reuse of cases expressed in textual form (Ashley, 1999). Process-embedded textual CBR that actively interjects lessons during decision making meets many KM requirements, including knowledge capture, storage, and reuse; knowledge growth, communication, and preservation, and knowledge gathering, structuring, refinement, and distribution (O'Leary, 1998; Decker and Maurer, 1999). Therefore, one of our focal objectives is to identify textual CBR techniques to support case *authoring* (*i.e.*, identifying, eliciting, representing, and indexing cases). Researchers have investigated using machine learning techniques (Brüninghaus and Ashley, 1999) to automate case authoring from unstructured texts, and template techniques for structured texts (Weber et al., 1998). Rose and Belew's (1991) approach for training semantic networks may also be applicable to indexing lessons. Latent semantic analysis (LSA) (Deerwester et al., 1990), a statistical method that compares the contents of texts, also shows potential in supporting indexing. According to Foltz et al. (1999), LSA can automate the process of essay scoring (i.e., by comparing the content in essays). It can also identify missing subtopics or structures. We are also considering using intermediate data structures to assist the case authoring process. For example, concept maps (Cañas et al., 1999) are an example of semi-structured representations that can enhance the text retrieval process. Independent of which textual CBR techniques we will select, we plan to use supporting tools (e.g., Worldnet) to construct an ontology for the domain to simplify similarity assessment. Information extraction and machine learning techniques (e.g., Riloff, 1993) may be useful for constructing these ontologies.

4. Active Lessons Learned in Plan Authoring

4.1 Conversational case-based plan authoring

Conversational case-based reasoning (CCBR) is an interactive form of CBR; it uses a mixed-initiative dialogue to guide users through a question-answering sequence to facilitate case retrieval (Aha and Breslow, 1997). This approach, introduced by Inference Corporation for customer support applications, was recently extended to decision support tasks in HICAP. HICAP assists a user with interactive plan elaboration and, under user control, passes the constructed plan to an execution monitor. HICAP has four modules:

1. *Hierarchical task editor* (HTE): This inputs the focal operation's doctrine and resources, represented as user-modifiable hierarchical task networks (HTNs).
2. *Conversational case retriever* (NaCoDAE/HTN): Users can select this module to interactively decompose a given task into subtasks. At any point during a "conversation" users can answer to a displayed question to improve the relevance of case solution rankings, or select a highly ranked case's solution to implement its task decomposition.
3. *Generative planner* (JSHOP): Users can also select this module to decompose tasks. JSHOP, a Java translation of SHOP (Nau et al., 1999), automates task decomposition for otherwise "tedious" tasks. This module requires a set of planning operators.
4. *Decision tracker* (DecTS): This module alerts users to resource conflicts in the emerging plan.

HICAP manages six objects: the hierarchical plan (including each task's duration), the resource hierarchy, assignments of resources to tasks, task relations (*e.g.*, temporal), the world state, and the set of existing conflicts.

4.2 Noncombatant evacuation operations

We will illustrate an active lessons delivery system in HICAP for planning NEOs, which are military operations for evacuating noncombatants, nonessential military personnel, selected host-nation citizens and third country nationals whose lives are in danger to an appropriate safe haven. They usually involve a swift insertion of a force, temporary occupation of an objective (*e.g.*, a USA Embassy), and then a planned withdrawal. NEOs are usually planned and conducted by a joint task force (JTF), and are under an Ambassador's authority. Force sizes can range into the hundreds; the evacuees can number into the thousands. At least ten NEOs were conducted during this decade (Siegel, 1995). Unclassified publications describe NEO doctrine, case studies, planning issues, and other types of general analyses. Please see www.aic.nrl.navy.mil/~aha/neos for links to references on unclassified NEO publications.

NEOs are challenging to plan; they require considering a wide range of factors, uncertainties (*e.g.*, hostility levels/locations), and hundreds of subtasks (*e.g.*, evacuee processing). Flawed plans can be disastrous. Therefore, NEOs are planned with the guidance of military doctrine (DoD, 1997), which addresses strategic and operational issues, but not most operation-specific tactical issues. Thus, to meet the demands of a specific NEO, the JTF commander (CJTF) (1) modifies the doctrine as needed and (2) employs experiences from previous NEOs, which complement doctrine by suggesting tactical refinements that are suitable for the current operation. For example, past experiences could help identify whether evacuees for a specific operation should be concentrated at an embassy or grouped at multiple evacuation sites.

Currently, HICAP allows NEO planners to benefit from previous NEO experiences that can be used to modify only the hierarchical NEO plan. However, NEO lessons can, more generally, be applied to update any of HICAP's six objects.

4.3 An active lessons delivery module

Cases corresponding to lessons learned experiences will be indexed by HICAP's objects, and their recommendations can affect any of these objects (*e.g.,* in HICAP: task insertion, edits to resource assignments). Using a CBR engine to compare HICAP's objects with the stored lessons, the most similar lessons will be proactively and automatically brought to the decision maker's attention. These "active" lessons perform a critiquing role; a lesson's recommendation explains how HICAP's objects should be changed for a given set of conditions. The user can either implement or ignore these recommendations. We are designing the active lessons delivery module so that it can implement recommendations automatically, but under user control. Figure 1 summarizes this approach.

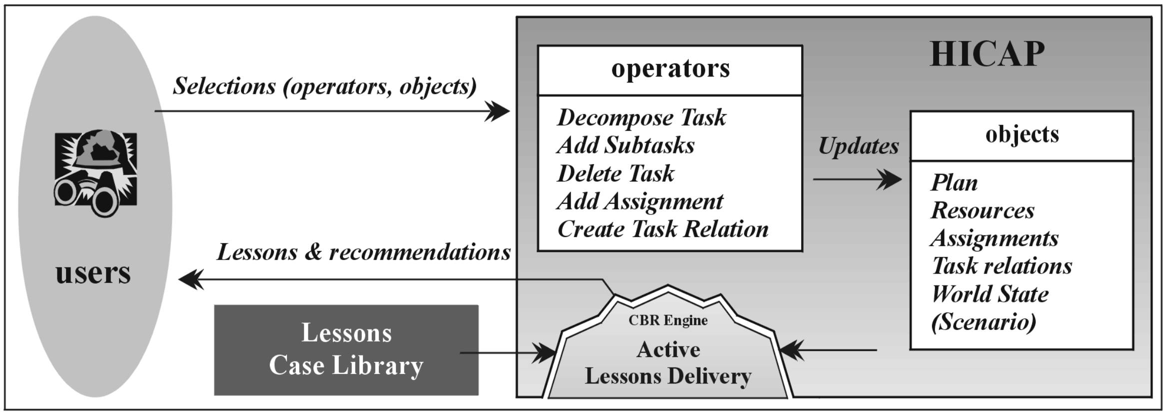

Figure 1: A Lessons Delivery Module in the HICAP Architecture.

4.4 Example

Active lessons can be tailored to different domains and decision support tools. Suppose HICAP is being used to help plan the location of an Intermediate Staging Base (ISB) in a NEO. ISB selection subtasks include coordination and composition tasks, among others (*e.g.*, coordinate with local security forces). Also, suppose a user decomposes an ISB selection task (*i.e.*, manually, via case application, or plan generation), yielding the subtask *coordinate with airfield traffic controllers*, and the world state is informed that it is a commercial airfield. Finally, suppose HICAP's lessons learned module is given a lesson (Figure 1) indexed by <ISB is a commercial airfield> with a recommendation to (1) add a subtask: assign military air traffic controllers and (2) add a subtask: transport military air traffic controllers to this ISB. Then the module would proactively alert the user with these recommendations.

UJTL Task: OP 4.5.1 Provide for Movement Services in Theater of Operations/JOA.
Context features: Evacuation airfield operation, rapid build-up of military flight operations, civilian airport is transformed into an intensive military operating area.
Lessons learned: Military flight operations overload civilian (host nation) controllers; military air traffic controllers are required whenever a civilian airport is used for military operations, and are critical for augmenting host country controllers to ensure safe evacuation airfield operations.
Recommended Actions: Ensure military air traffic controllers are part of the evacuation package or assign military air traffic controllers to the evacuation package.

Figure 2: Some Attributes of a Lesson.

The decision maker can choose to implement the lesson's recommendations, a subset of them, or ignore them. The recommendation's interpretation will be provided by the module (or will be stated clearly by the lesson representation), but can be overridden by the decision maker. In summary, an active lessons delivery module can contribute additional knowledge to the planning task, outside the scope of doctrine, based on the experience of other decision makers who performed similar tasks. This facilitates the reuse of knowledge that was once individual, now making it available as organizational knowledge.

5. Future and Related Work

The JCLL publishes calls for contributions to the joint community that require the thesis of a lesson, but not in a sophisticated format. Therefore, an interactive elicitation module, perhaps similar to HICAP, could assist with eliciting lessons from decision makers. This could be complemented by a module for eliciting *concept maps* (Cañas et al., 1999), which would simplify lesson communication. Both modules could incorporate ambiguity resolution procedures. This strategy would simplify the lesson acquisition task, freeing lessons learned specialists to focus on other aspects of lessons learned systems (*e.g.*, embedding them in decision support tools).

We plan to use NaCoDAE (Aha and Breslow, 1997) to assist with lesson elicitation, which could assist the military services in becoming learning organizations, from the perspective of sharing lessons learned repositories. Some alternatives for representing lessons learned as cases were discussed in Section 3. The selected representation should be adaptable, easily maintained, and maximize recall and precision while retrieving relevant similar experiences.

We will also convert existing repositories to this representation so that they can be retrieved using a CCBR engine. This requires determining each lesson's relevant features. We will identify an ideal textual representation as a starting point for these lessons. Since lessons are stereotypical in nature, rewriting lessons text will allow us to use template techniques combined with rhetorical structures (Weber et al., 1998). Thus, we can engineer existing lessons into these text structures and use this experience to define elicitation guidelines.

Although two workshops have recently taken place on lessons learned processes (Secchi, 1999; SELLS, 1999), neither targets the use of AI for lessons learned systems.

Acknowledgements

Many thanks to Kevin Ashley for his feedback on this paper. Thanks also to Len Breslow and Héctor Muñoz-Avila for consultations on HICAP. This research was sponsored by grants from the Office of Naval Research, Naval Research Laboratory, and the University of Wyoming.

References

Aha, D.W., and Breslow, L.A. 1997. Refining conversational case libraries. In *Proceedings of the Second International Conference on Case-Based Reasoning*, 267-278. Providence, RI: Springer.

Aha, D.W., Becerra-Fernandez, I., Maurer, F., and Muñoz-Avila, H. (Eds.). 1999. *Exploring Synergies of Knowledge Management and Case-Based Reasoning: Papers from the AAAI 1999 Workshop*, Technical Report, WS-99-10. Menlo Park, CA: AAAI Press.

Ashley, K. 1999. *Progress in Text-Based Case-Based Reasoning*. Invited talk, 1999 International Conference on CBR. Available: www.lrdc.pitt.edu/Ashley/TalkOverheads.htm

Becerra-Fernandez, I., and Aha, D.W. 1998 Case-based problem solving for knowledge management systems. In *Proceedings of the Twelvth Annual Conference of the International Florida Artificial Intelligence Research Society*, 219-223. Orlando, FL: AAAI Press.

Brüninghaus, S., and Ashley, K.D. 1999 Bootstrapping case base development with annotated case summaries. In *Proceedings of the Third International Conference on Case-Based Reasoning*, 303-314. Seeon, Germany: Springer.

Cañas, A., Leake, D., and Wilson, D.C. 1999. Managing, mapping, and manipulating conceptual knowledge. In Aha et al. 1999, 10-14.

Davenport, T.H., and Prusak, L. 1998. *Working knowledge: How organizations manage what they know*. Boston, MA: Harvard Business School Press.

Decker, S., and Maurer, F. eds. 1999. Organizational memory and knowledge management. Special issue in *International Journal of Human-Computer Studies 51*(3): *511-714.*

Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., and Harshman, R. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science 41:* 391-407.

DoD 1997. *Joint tactics, techniques and procedures for noncombatant evacuation operations*. (Joint Report 3-07.5). Washington, DC: Department of Defense.

Foltz, P.W., Laham, D., and Landauer, T.K. 1999. Automated essay scoring: Applications to educational technology. In *Proceedings of EdMedia' 99*. Retrieved 11-05-99 from: http://www-psych.nmsu.edu/~pfoltz/reprints/Edmedia99.html

Muñoz-Avila, H., McFarlane, D., Aha, D.W., Ballas, J., Breslow, L.A., and Nau, D. 1999. Using guidelines to constrain interactive case-based HTN planning. In *Proceedings of the Third International Conference on Case-Based Reasoning*, 288-302. Munich: Springer.

Nau, D., Cao, Y., Lotem, A., and Muñoz-Avila, H. 1999. SHOP: Simple Hierarchical Ordered Planner. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 968-973. Stockholm, Sweden: Morgan Kaufmann.

O'Leary, D. 1998. Enterprise knowledge management. *Computer*, *31*(3): 54-61.

Riloff, E. 1993. Automatically constructing a dictionary for information extraction tasks. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, 811-816. Menlo Park, CA: AAAI Press.

Rose, D.E., and Belew, R.K. 1991. A connectionist and symbolic hybrid for improving legal research. *International Journal on Man-Machine Studies, 35:* 1-33.

Sary, C. and Mackey, W. 1995. A case-based reasoning approach for the access and reuse of lessons learned. In Proceedings of the Fifth Annual International Symposium of the National Council on Systems Engineering, 1:249-256. St. Louis, Missouri: NCOSE.

Secchi, P. ed. 1999. Proceedings of Alerts and Lessons Learned: An Effective way to prevent failures and problems, Technical Report, WPP-167, Noordwijk, The Netherlands: ESTEC/ESA.

SELLS. 1999. *Proceedings of the Society for Effective Lessons Learned Sharing.* (SELLS) Spring Meeting. Las Vegas, NV: Unpublished. Retrieved 01-06-00 from: www.tis.eh.doe.gov/ll/sells/proceedings399.htm

Senge, P. 1990. *The fifth discipline: The art and practice of the learning organization*. London: Random House.

Siegel, A. 1995. *Requirements for humanitarian assistance and peace operations: Insights from seven case studies*, Technical Report, CRM 94-74. Arlington, VA: Center for Naval Analyses.

Watson, I. 1997. *Applying case-based reasoning: Techniques for enterprise systems*. San Francisco, CA: Morgan Kaufmann.

Weber, R. 1999. Intelligent jurisprudence research. In *Proceedings of the Seventh International Conference on Artificial Intelligence and Law*, 164-172. Oslo, Norway: ACM.

Weber, R., Martins, A., and Barcia, R. 1998. On legal texts and cases. In M. Lenz and K. Ashley eds. *Textual Case-Based Reasoning: Papers from the AAAI-98 Workshop,* 40-50. Technical Report, WS-98-12. Menlo Park, CA: AAAI Press.