Title
Early Validation of Task Analysis Data: Processes and Representations

Permalink
https://escholarship.org/uc/item/3q31f2w8

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 20(0)

Authors
Haddock, Gail
Priest, John
Harbison, Karan
et al.

Publication Date
1998

Peer reviewed

Early Validation of Task Analysis Data: Processes and Representations

Gail Haddock (haddock@caesar.org)
John Priest (priest@caesar.org)
Karan Harbison (harbison@caesar.org)
Center for Advanced Engineering and Systems Automation Research
The University of Texas at Arlington, PO Box 19066, Arlington, TX 76019-0066

John Silva (jsilva@darpa.mil)
Defense Advanced Research Projects Agency (DARPA)
Defense Sciences Office, 3701 North Fairfax Drive, Arlington, VA 22203-1714

Abstract

Task analysis is a critical first step in understanding a new complex domain. Currently, task analysis is a mostly manual process with weak automation support. This paper introduces the first phase of the SAVVII prototype as a proof-of-concept for early validation of task analysis activities. Early validation is supported by the transference of semantics from data values to data structures. Rough estimations of discrepancies between tasks are used to focus the knowledge elicitor's attention on questionable areas, thereby reducing much of the tediousness and time-intensive nature of validation. SAVVII was shown to work on the developmental domain of parables. It is currently undergoing experimentation in two real-world knowledge acquisition activities.

Introduction

Many systems engineering methodologies call for extensive efforts in knowledge acquisition along with design, implementation, and test. Task analysis is a critical phase in many knowledge acquisition processes. To ensure the quality of task analysis results, validation is often performed by comparing the results with additional sources of information. This validation process is currently manual, tedious, time-consuming and expensive. However, using possibly erroneous task analysis results accepted from a weak validation process inserts too much risk into new system development.

Current work in applied ontologies is one approach to reducing task analysis risk. In practice, limited financial resources often result in a lack of proper effort given to the ontology creation process. This is due to the mostly manual process of ontology creation, requiring a collaborative effort among many uniquely talented individuals. Automated tool support for both the humanistic nature of knowledge acquisition and the exacting nature of ontology formalization is needed.

The objective of this research is to develop a process and representation framework that will facilitate partial automation of the validation process. Requirements of a task analysis validation framework include:

- support validation as early as possible in the knowledge acquisition process;
- support timely notification of events requiring intervention by knowledge acquisition participants;
- provide validation in the form of the identification of discrepancies between newly entered task analysis data and previously validated data; and
- adapt to validation biases of the project, domain, knowledge elicitor, or ontology system.

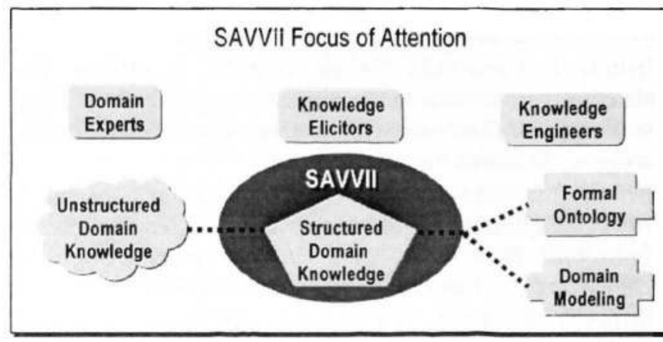

Figure 1: SAVVII is a tool intended for knowledge elicitors as an information conduit from domain experts to knowledge engineers. By adding only partial structure to otherwise unstructured domain knowledge, validation can occur earlier in the lifecycle.

The System for Assisting with the Verification and Validation of Imperfect Information (SAVVII) is a prototype designed to achieve the research objectives and function as an aid to knowledge elicitors as they translate knowledge from domain experts to knowledge engineers, as shown in Figure 1. It is a "System for Assisting" in that the user retains the ultimate responsibility for final determination of "Verification and Validation" activities, which is supported through an automated process of similarity and discrepancy determination followed by feedback from domain experts. "Imperfect Information" describes the evolutionary state of task data as the core representation. All task data originally appears as a potentially erroneous perspective of the domain. Through assimilation, conflict resolution, validation, acceptance, and management of this imperfect knowledge, task data evolves into information that is usable by other lifecycle systems. Future work will expand the focus of SAVVII from task analysis to other types of knowledge engineering processes, such as interaction, concept, and decision analyses.

The first phase of the SAVVII prototype has focused on a short validation cycle for the knowledge elicitor. The processes and representations of this phase are presented in this paper after a brief review of task analysis concepts. Future phases of SAVVII development will introduce assimilation processes, acquisition of multiple domain perspectives, and visualization and manipulation of task data through visualization mechanisms.

Task Analysis

Task analysis as part of domain analysis is designed to elicit detailed descriptions of domain tasks. Task analysis has evolved in separate communities. Each community has developed task analysis techniques that best fit their requirements.

Industrial engineering is concerned with operator capability (Kirwan, & Ainsworth, 1992). From an industrial engineering perspective, task analysis is useful in the areas of system design (safety, productivity, and system availability), system evaluation (internal company audits, risk assessments), and specific concerns (technology vulnerable to human error, system changes causing uncertainty about system integrity, implementation of new constraints). Team effectiveness is the central focus of task analysis in organizational management. The goal is to understand the relationships between teams, tasks, and organizational structures such that team performance can be predicted (Cannon-Bowers, et al., 1995).

Understanding a process such that new automation systems can be integrated into specific problem areas is the focus of task analysis in software engineering. Tasks are studied so that new systems can be built that will either perform those tasks for the users or support the users in their accomplishment of the tasks. Tasks are decomposed such that behavior of the process is apparent (McGraw & Harbison-Briggs, 1989). Behaviors of actors who currently perform the task are significant with respect to what they do, rather than what they know. (Other knowledge acquisition techniques are useful in eliciting what an expert knows, and are not within the scope of this paper.) An additional area of concern is the task's relationship to the goals of the whole system.

Knowledge engineering is focused on understanding a domain, with the end-use of the knowledge of less a concern than the other disciplines previously discussed. Knowledge engineering is an attempt at combining all other perspectives. Task analysis is used to understand a domain through constraining its description into formalized model representations (Arango, 1993). Modeling limits the description and limits the complexity, so that various analysts can draw their own conclusions by studying the models rather than be overwhelmed by unconstrained data.

Perhaps the simplest model is the task template. Table 1 lists task decomposition fields of a task template (Kirwan & Ainsworth, 1992). The template allows the knowledge engineer to gather as much information in as many different categories as possible through one set of knowledge acquisition sessions with experts.

Table 1: Task Template: descriptive decomposition categories. Adapted from Kirwan and Ainsworth (1992).

Description	Description	**Hardware**	Location
	Type of activity or behavior		Controls used
	Task or action verb		Displays used
	Function or purpose		Critical values
	Sequence of activity		Job aids required
Requirements	Initiating cue or event	**Performance**	Performance metric
	Information		Time taken / starting time
	Skills / training required		Required speed
	Personnel required / staffing		Required accuracy
Nature of task	Action required		Criterion of response adequacy
	Decisions required	**Other items**	Subtasks
	Responses required		Communications
	Complexity		Coordination requirements
	Task difficulty		Concurrent tasks
	Task criticality	**Consequences**	Likely / typical errors
	Amount of attention required		Problems
Outputs	Output from the task		Error consequences
	Feedback		Adverse conditions / hazards

Task Terminology

The ability to map between different ontological concepts that might result from task analysis resides in the ability to understand and manipulate terminology. The terminologist builds a terminological database as a list of terms with associated linguistic information and a natural language definition (Aussenac, 1995). Linguistic analysis of text usually consists of identification of terms and their semantic relations, followed by extrapolation of concepts from those terms. This data structure is insufficient for conceptual understanding required by knowledge engineers.

Terminologists focus almost entirely on statistical text analysis processes. The text is considered the ultimate reference, as it is assumed to be the result of a consensus of domain experts. The SAVVII prototype facilitates a strong tie to terminology fidelity and depends on that fidelity for its validation processes.

Task Model

The data representation of this research is based on the CommonKADS Task Model (Duursma & Olsson, 1994). For a historical perspective of KADS, CommonKADS, and ESPRIT, see (Schreiber & Wielingua, 1996), where CommonKADS is introduced as "the most widely used method for KBS development." Concepts introduced in this section are from McGraw and Harbison-Briggs (1989) for task information and Duursma and Olsson (1994) for task modeling.

An *activity* refers to actions performed in the real world. A *task* is a set of coherent activities that are performed to achieve a goal. Task descriptions add a teleological dimension, such as the purpose for performing the task. The *task model* is a resulting artifact of task analysis. The task model consists of functions of an organization in terms of tasks. A *scenario* is a description of a sequence of tasks performed as one thread of execution in the domain. A scenario contains an ordered set of tasks.

A *process model* is the collection of all tasks performed in all domain-relevant scenarios, along with the rationalization for selecting subsequent tasks as indicated in the scenarios. Therefore, the process model adds sequencing and timing decision points that guide an actor in performance of particular tasks in a given context. However, the task model is not concerned with organizational processes. The focus is on the performance of the task itself in light of the environmental context, but not with subsequent activities in a given scenario, or task interaction in a particular process. Therefore, the task model consists of an unordered set of tasks.

The CommonKADS Task Model is highly integrated with external CommonKADS models, including Design, Agent, Communication, Expertise, and Organization Models. These external models are not the subject of this research. However, it is expected that a robust application should also be able to reason sufficiently over these external representations, and is the subject of future efforts.

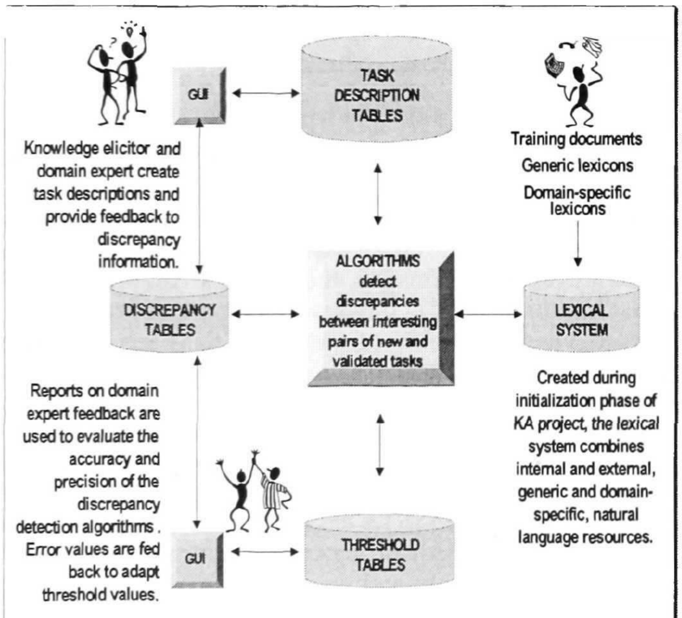

Figure 2: Component-based Architecture of SAVVII.

Validation Representations and Processes

The SAVVII prototype has been implemented using commercial-off-the-shelf components and publicly available components. Microsoft Access 97 was chosen as the database component due to its relative affordability, laptop capability, and replication facility. The replication facility is important for traveling knowledge elicitors. Each elicitor can add data to the database while in the field. Through a modem connection, the fielded databases can be periodically synchronized with a master database. Only changes to the databases are transmitted. During the synchronization process, each elicitor in the field receives task data from all previously synchronized field elicitors, providing a validation activity that is current even in a distributed operational environment.

Representations

SAVVII database components consist of tables for task descriptions, discrepancies and thresholds, and a lexicon taxonomy for semantic comparison (see Figure 2).

Task Description Tables The SAVVII Task Model is an implementation of a specialized version of the CommonKADS Task Model. It is a relational database with intermediary tables that support the many-to-many relations required between satellite tables and the entry-point task table. The many-to-many relations help to reduce size of the database by supporting only one entry for repeated instances among tasks. For example, one task may be performed by many agents, and one agent may perform many tasks. Each agent is entered only once, just as each task is entered only once. Only the pointers to these entries are repeated in the relation tables.

Figure 3 shows the satellite tables related to the task table through many-to-many relation tables. These tables are drawn mostly from the CommonKADS Task Model, with the inclusion of particular tables found to be useful in other knowledge acquisition projects. For example, both the Hazards and Errors tables were subsumed by the Features table of the CommonKADS Task Model. However, an increase in specificity of the satellite tables improves the comparative capability within each table. More accuracy is expected when comparing hazards to hazards and errors to errors, than is expected if hazards and errors were intermixed in the same table. Upon continued use of SAVVII, more satellite tables may be defined and populated depending on the context of the active domain. Table creation is guided by an ontology that specifies concepts as significant for task analysis. Expansion of SAVVII into other analyses of knowledge engineering depends on the creation of ontologies to guide appropriate table creation.

Early validation depends on the ability to transfer semantic information from raw KA notes to ontological categories. Ontological categories are represented as satellite tables in the prototype. The knowledge elicitor is expected to perform the semantic transfer by assigning segments of the text to a category. Other segments of text are ignored because they provide the same semantic information as the category. Therefore, semantic information is transferred from the text to the category. When selecting a previously existing satellite table entry for a new task, the knowledge elicitor is classifying the entry as similar for both tasks. The specificity of category creation guided by the ontology is contrasted against the cognitive ability of the knowledge elicitor to perform similarity classification during data entry. Further research into this tradeoff is expected. An increase in the number of categories results in an increase in the semantic transfer from data to structure. However, a corresponding increase in the complexity of data entry upon the knowledge elicitor at some point may result in cognitive overload and result in error insertions or disuse.

Not shown in Figure 3 are the metamodel tables. These tables are used to track the knowledge acquisition activity. Information provided in each task model is associated with the expert who provided the information, the type of knowledge elicitation technique used to acquire the knowledge, the source document from the technique (or written document if no expert was used), information on how to access the source document (storage location and medium, etc.) and the knowledge engineer or elicitor involved in the process. This metamodel data is available for future quality determination experiments, such as the distribution of contributions over experts and novices per organization.

Discrepancy and Threshold Tables Separate sets of data tables are maintained for discrepancy filtering. Task data that has not yet passed the filtering process are analyzed for similarities and discrepancies with task data previously accepted. Questionable data remains in these discrepancy tables as persistent storage until such a time that a knowledge engineer rejects, modifies, or accepts the questionable data into the Task Model Database. The acceptance of questionable data must be accompanied by a rationale for its acceptance. This rationale is maintained with the data for traceability.

Threshold tables are used in SAVVII to apply numerical values as bounds for discrepancy classification. These values are manipulated during experimentation and are stored for future analysis. Weighting factors per satellite table are also stored with the thresholds. Weighting factors facilitate a domain bias during similarity comparison.

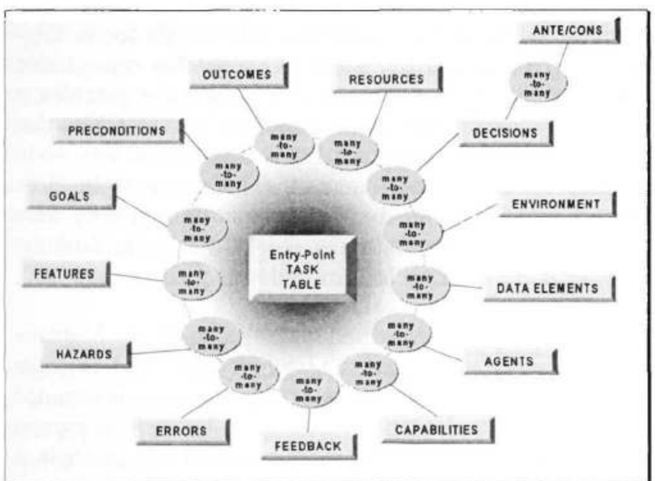

Figure 3: Task Model Relational Tables. Semantics of the relation is maintained in the structure of the tables.

Processes

For the SAVVI prototype, algorithms were implemented in Visual Basic for Applications due to its close integration with Microsoft Access 97. Algorithms were designed for proof-of-concept motivations, and to be easily replaced with sophisticated techniques after concept maturation. Processes concern lexicon maintenance, statistical metrics, discrepancy determination, and validation feedback.

Lexical System At the core of discrepancy determination lies semantic comparison. Semantic comparison in turn relies on a solid terminological base. Existing terminology tools specializing in segments of the English language can be integrated into the SAVVII framework. One such tool is WordNet, developed at Princeton University (Miller, 1995). WordNet is a hub-based taxonomy of common English terms. Another tool is the Library of Medicine's Unified Medical Language System (UMLS), a medical terminology taxonomy (Lindberg, Humphreys, & McCray, 1993). Specialty taxonomies for jargon-rich applications like healthcare can be integrated into SAVVII through the same mechanisms implemented for common English terminology. Internal to SAVVII is a lexicon that is used to maintain a taxonomy of terms particular to the SAVVII database, and not available in the associated external terminology taxonomies. Terms that are not found in the external taxonomies are entered into the table. Evaluation of the table is supported, including rating its cohesiveness and viability. Users of the system are encouraged to enrich the lexicon as new terms are discovered.

Statistical Metrics Statistical metrics are concerned with identification of additional relationship patterns in the task model. These relationships include the level of relative contribution by experts, documents, and knowledge engineers. The distribution of contribution per expert, for example, is readily identifiable by analyzing the tables that facilitate the many-to-many relations between tasks and experts.

This statistical data can be associated with discrepancy data. The result could be useful in analyzing the quality of the contributing expert, knowledge engineer, or document. For example, one expert may contribute data that repeatedly results in discrepancies with many otherwise consistent experts. Detailed correlation may show that this expert's contributions should be considered suspect. At the extreme, all data contributed by this expert could be marked questionable, or even eliminated entirely from the knowledge base. Radically different information does not always point to errors, as this expert may be the only sane voice crying out in the darkness. The inclusion of a feedback loop allows a user to override the computer's determination of discrepancy. Radically different data can remain in the knowledge base and serve as an indicator to the knowledge elicitor that more knowledge acquisition is necessary. In the future, sophisticated technologies for data mining will enhance the capabilities of such analyses in complex domains.

Discrepancy Determination One of the more complex, imprecise, yet critical functions of the task analysis validation is the automatic classification of similar and discrepant data. Similar tasks are difficult to assess in unrestricted data collection processes. A rough similarity rating can be calculated between two tasks through analysis of the structural semantics resident in the relation tables. For example, if Task A and Task B share more than some threshold percentage of the exact same satellite table pointers (same agents, same errors, same decisions, etc.), then those two tasks are considered similar enough to promote further investigation. If they share less than the threshold, then they are not interesting enough to warrant further study.

Initially, the rating is based on a simple percentage adjusted by a weighting factor. The weighting factor facilitates a satellite table bias. Further research is required as to the appropriateness of the threshold value, as well as more sophisticated similarity formulas. The threshold can be automatically adjusted during analysis feedback stages. One might hypothesize that different applications will adjust this threshold to different levels. An interesting future study could determine if the domains themselves cause the difference, or if the value depends more on the individual users.

Once two tasks have been classified as similar, further analysis of their differences is attempted. This analysis requires an in-depth look into the contents of the records that do not match in each task's satellite table. The assumption is that there is some discrepancy. Otherwise, users would have selected pointers to identical records. The extent to which the discrepancy is significant can be measured through semantic distance evaluations.

Several researchers have implemented semantic distance evaluations in comparing text fields, for example (Smeaton & Quigley, 1996; Wilensky, 1993). In SAVVII, terms within the fields are compared first for synonym matching and then for hypernym distancing. The values are accessed from the systems lexicon or an external term taxonomy. Synonym matching provides a level 1 rating of word distance. Two tasks whose only difference was the use of synonyms would see a reduction or even an elimination of the discrepancy classification.

Another form of semantic distance can be calculated when two terms share an accessibility path in a hierarchy created by overlapping their hypernym paths. For example, the terms "controller" and "medic" may meet in the hierarchy at the point of "human." However, this meeting may occur at several levels above the individual terms. A threshold parameter governs when two terms are too far apart to be considered similar. User feedback can affect the value of this threshold such that the system performs better through extended use. Due to the component framework structure of the architecture, semantic distance algorithms that perform better than the current approach can be readily integrated into the system.

To speed up computation, words are paired based on their part-of-speech. Word pairs and their calculated semantic distance are stored in a database table. This table is accessed through a lookup function to provide a quick response rather than initiate a detailed analysis on previously compared terms. A study will be made to determine the effectiveness and limitations of scaling up this lookup table for large complex domains.

Validation Feedback When discrepancies are detected, the new task is not assimilated into the accepted task model. Its discrepancy information is transferred to the discrepancy database until feedback is received from the knowledge elicitor. Feedback can be of several forms. Experts might indicate a correction is needed due to a typographical or semantic error. Some experts, when facing conflicting opinions of other experts, might change their mind. New information may be added to the task data, such as a particular context that must hold for the task data to be accurate. They may indicate that the data they provided is correct, and the data in the previously accepted task is in error. Experts might also indicate that the determination of the discrepancy is incorrect.

Experimental Domains

Biblical parables were used for SAVVII's development domain. Parables were chosen due to their resemblance to scenarios gathered during typical KA sessions, their jargonless word selection, and their familiarity to many English readers.

> *"A parable is an extended metaphor or illustration. ... Like most illustrations, parables usually have one central point. They differ from allegories, stories in which the details abound with hidden meanings. Failure to distinguish between parables and allegories, and the attempt to treat the parables of Jesus as though they were allegories, has led some Bible students far afield, pursuing mysterious meanings in meaningless details."*
>
> *John White (1988)*

White's concept of "meaningless details" is the rationalization for this research's efforts of rough classification of "interesting" tasks, rather than precise semantic understanding as is necessary for allegorical reasoning.

Two parables in the following explanation are considered by many to be the same parable told by different authors. They both deal with error conditions that result from scattering seed while sowing. It is relatively simple for an English-reading person to immediately judge that one parable validates the other. However, after reading many parables, it may become more difficult to recall all past parables that are similar in some aspects yet different in others.

Parables from the New International Version of the Holy Bible were entered into the database as though they were generic farming tasks. As many parables concern farming, they have some overlap in terminology, such as seed, soil, and sowing.

SAVVII properly classified parables such as Matthew 13:24-30 as being uninteresting when compared to Luke 8:5-8. However, Matthew 13:3-8 was not only determined to be interestingly similar to Luke 8:5-8, but certain aspects were identified as warranting more attention. An example is the comparison between the crop being "a hundred, sixty or thirty times what was sown" verses "a hundred times more than was sown". The question is not only which one is correct, but is this difference significant or meaningless, can they be unified into "30 to 100 times", or does one preclude the other. Discrepancies such as these require feedback from an expert through further interviews with the knowledge elicitor. The ability of the system to focus the attention of users on only those areas that warrant such attention is intended to reduce much of the tediousness of manual validation.

At this time, two experiments are being conducted to test the capabilities of the SAVVII prototype in real-world KA applications. Aircraft scheduling anomaly resolution is the domain of the first experiment, chosen due to its low degree of special jargon, limited degree of complexity, and high degree of undocumented expertise. The oncology medical domain was chosen as the domain second for the second experiment due to its heavy reliance on specialized vocabulary, extreme complexity of unknown factors, and high degree of erroneous data.

Future Work

The SAVVII prototype has proven the value of early validation. Much multidisciplinary work remains to be done towards theoretical foundations and specific technical capabilities. Current functionality of the *user interface* will not scale up for large complex domains. New selection processes will need to be explored to facilitate the cognitive ease of which SAVVII usage depends. Automatic detection of new *terminology* during data entry would make it easier for the knowledge elicitor to realize when the system is introduced to a new term. The ability to define the term at the point of entry also helps to ensure the person defining the term does so under the context of its usage.

As research in *natural language resources* continues to produce advances, these systems can be integrated into the algorithms. This would result in a continuous enhancement to the accuracy of the analysis process. The mechanisms used for *similarity comparison* in this research are unsophisticated at best. Further analysis and integration of sophisticated similarity methods may result in using more efficient comparison techniques, such as graph analogy reasoning, decision theory, or case-based reasoning.

The ability to view the task hierarchy is currently limited to the supertask/subtask taxonomy. Other views might be more appropriate for given situations and will require support from *sophisticated search* mechanisms with automated taxonomy creation capabilities. A mechanism for *the unification and specialization* of near neighbor tasks is key to converting validated data into information useful to subsequent lifecycle systems, such as assimilation based on context, feedback, terminology or similarity.

Conclusion

Task analysis results are a critical first step in understanding a new complex domain. Currently, task analysis is a mostly manual process with weak automation support. This paper introduces the first phase of the SAVVII prototype as a proof-of-concept for early validation of task analysis activities through discussion of validation processes and representations. SAVVII was shown to work on the developmental domain of parables. It is currently undergoing experimentation in two real-world KA activities. Future phases of SAVVII development will enhance the prototype with sophisticated technologies that will lead to a robust knowledge representation system linking task relevant data with multiple domain knowledge sources.

Acknowledgments

This article is dedicated to the memory of one of the authors, Dr. Karan Harbison, who passed away January 3rd after a short battle against cancer. Funding for this effort was provided by the Defense Advanced Research Projects Agency (DARPA) Intelligence Information Infrastructure (III) Program's Environment for Advanced Engineering Systems Environment (EASE) project, DARPA Clinical Associates (CLASS) Program's Scenario-based Engineering (SEP) project, the National Science Foundation, the Texas Higher Education Coordinating Board (THECB) Advanced Technology Program (ATP), and Systems and Processes Engineering Corporation (SPEC).

References

Arango, G. (1993). Domain Analysis Methods. Austin, TX: Schlumberger Laboratory for Computer Science.

Aussenac, N. (1995). How Can Knowledge Acquisition Benefit from Terminology? *Proceedings of the 9th Knowledge Acquisition for Knowledge-Based Systems Workshop*. Banff, CA.

Cannon-Bowers, J.A., Tannenbaum, S.I., Salas, E., & Volpe, C.E. (1995). Defining Competencies and Establishing Team Training Requirements. In R.A. Guzzo, E. Salas, and Assoc. (Eds.), *Team Effectiveness and Decision Making in Organizations*, San Francisco, CA: Jossey-Bass.

Duursma, C. & Olsson O. (1994). *Task Model Definition and Task Analysis Process*.(Document Id: KADS-II/M5/VUB/RR/004/2.0). Brussels: Vrije University.

Holy Bible, New International Version, Colorado Springs, CO: International Bible Society.

Kirwan, B., & Ainsworth L.K. (Eds.) (1992). *A Guide to Task Analysis*. London, UK: Taylor & Francis.

Lindberg, D.A.B., Humphreys, B.L., & McCray A.T. (1993). The Unified Medical Language System. *Methods of Information in Medicine*, 32, 281-291.

McGraw, K. & Harbison-Briggs, K. (1989). *Knowledge Acquisition: Principles and Guidelines*. Englewood Cliffs, NJ: Prentice Hall.

Miller, G.A. (1995). WordNet: A Lexical Database for English. *Communications of the ACM*, v38, n12, 39-41.

Schreiber, G. & Wielingua, B. (1996). SWI: making knowledge technology work. *IEEE Expert: Intelligent Systems and Their Applications*, v11, n2, 74-76.

Smeaton, A.F. & Quigley, I. (1996). Experiments on Using Semantic Distances Between Words in Image Caption Retrieval. *Proceedings of the 19th International Conference on Research and Development in Information Retrieval*. Zurich, SW.

White, J. (1988). *Parables: The Greatest Stories Ever Told*. Downers Grove, IL: InterVarsity Press.

Wilensky, R. (1993). Knowledge Acquisition and Natural Language Processing. In S. Chapman & A.L. Meyrowitz, (Eds.), *Foundations of Knowledge Acquisition: Cognitive Models of Complex Learning*, Boston: MA, Kluwer Academic Publishers.