Proceedings *of the* Thirteenth National Conference on Artificial Intelligence and the

Eighth Innovative Applications of Artificial Intelligence Conference

Volume One

Sponsored by the American Association for Artificial Intelligence

AAAI Press / The MIT Press

Menlo Park • Cambridge • London

KI: A Tool for Knowledge Integration

Kenneth S. Murray
Cycorp
3500 West Balcones Center Drive
Austin, Texas 78759
murray@cyc.com

Abstract

Knowledge integration is the process of incorporating new information into a body of existing knowledge. It involves determining how new and existing knowledge interact and how existing knowledge should be modified to accommodate the new information. KI is a machine learning program that performs knowledge integration. Through actively investigating the interaction of new information with existing knowledge KI is capable of detecting and exploiting a variety of diverse learning opportunities during a single learning episode. Empirical evaluation suggests that KI provides significant assistance to knowledge engineers while integrating new information into a large knowledge base.

Introduction

Knowledge integration is the process of incorporating new information into a body of existing knowledge. It involves determining how new and existing knowledge interact and how existing knowledge should be modified to accommodate the new information. Knowledge integration differs significantly from traditional approaches to learning from instruction because no specific performance task is assumed. Consequently, the learning system must assess the significance of new information to determine how existing knowledge should be modified to accommodate it. Typically, new information has many consequences for existing knowledge, and many learning opportunities will result from one learning episode.

This paper describes exploratory research that investigates knowledge integration in the context of building knowledge-based systems. The goals of this research include formalizing knowledge integration as a machine learning task, developing a computational model for performing knowledge integration, and instantiating the computational model in KI, an implemented machine learning program (Murray 1995). Specifically, this paper describes how KI performs knowledge integration and illustrates how KI exploits multiple learning opportunities during a single learning episode.

Knowledge integration addresses several critical issues that arise when developing knowledge bases. It is important to assess how new information interacts with existing knowledge because knowledge base modifications that are intended to correct a shortcoming may conflict with existing knowledge and introduce problems. For example, extending a drug therapy advisor (e.g., Mycin) to minimize the number of drugs prescribed to each patient conflicts with the therapy goal of maximizing the number of symptoms covered by the prescribed treatment (Mostow & Swartout 1986). Detecting and adjudicating conflicts as new information is entered prevents subsequent problem solving failures.

Alternatively, new information may interact synergistically with existing knowledge. One common type of beneficial interaction occurs when the new information explains existing knowledge. For example, adding the new information that chloroplasts contain the green pigment chlorophyll to a botanical knowledge base helps to explain the existing default beliefs that the leaves of plants are green and capable of conducting photosynthesis (Murray 1990). Recognizing this interaction between new and prior knowledge permits the system to better explain its conclusions.

Figure 1 presents a learning scenario that exemplifies learning as knowledge integration. The teacher presents new information about the anatomy of a plant leaf. The idealized student is very reactive and responds with several observations on the physiological effects of the leaf cuticle covering the leaf's epidermis. These responses indicate the student acquires knowledge beyond the literal content of the new information. For example, the student suggests generalizing the new information: not only does the leaf have a cuticle, but so do the other parts of the plant's shoot system. Furthermore, the student's responses reveal to the teacher the existing state of the student's knowledge, enabling the teacher to provide follow-up comments that resolve the student's questions and misconceptions. Significantly, the student is not limited to a single learning goal, such as defining a particular target concept, but instead acquires a wealth of diverse knowledge during this single learning episode.

Teacher: The epidermis of the plant leaf is covered by the leaf cuticle, which is composed of cutin.

Student: Cutin is impermeable to gases, so the cuticle restricts water loss from the leaf.

Teacher: Yes, that's right.

Student: By reducing water loss, the cuticle helps the leaf avoid dehydration. Other plant organs that transpire would also benefit from a cuticle. Do stems, fruits, and flowers have cuticles?

Teacher: Yes.

Student: But the cuticle also causes the leaf to starve.

Teacher: Explain.

Student: The cuticle is impermeable to gases. This prevents carbon dioxide in the atmosphere from passing through the leaf's epidermis. Without carbon dioxide, the leaf cannot conduct photosynthesis and starves.

Teacher: Well, the cuticle is impermeable to carbon dioxide; however, the leaf survives.

Student: Does the cuticle only partially cover the epidermis? Or, are there portals in the epidermis that permit restricted gas flow?

Teacher: Yes, the epidermis has portals called stomata.

Figure 1: Learning about leaf cuticle

As illustrated in Figure 1, learning proceeds without assuming how acquired knowledge will eventually be used. Consequently, the learning method cannot be tailored to a particular performance task, such as classification. Furthermore, the new information includes general rules. Thus knowledge integration differs from traditional machine learning tasks in two respects. First, the learning system cannot assume that new information conforms to a narrow format and content, such as classified instances of a target concept. Second, the significance of new information is often varied and hidden; therefore, the learning system must assess the significance of new information to determine how existing knowledge should be modified to accommodate it. When new information has many significant consequences for existing knowledge, many learning opportunities will result from the single learning episode. The following sections describe how KI exhibits the learning behavior presented in Figure 1.

KI: An Overview

KI is an interactive tool for knowledge integration. It was developed to help extend the Botany Knowledge Base, a large knowledge base representing plant anatomy, physiology, and development (Porter *et al.* 1988). The knowledge base contains about 4000 terms and 17,000 rules. However, it was implemented in a version (circa 1990) of the Cyc knowledge base (Lenat & Guha 1990), which contains about 27,000 terms and 40,000 rules, all of which are accessible to KI.

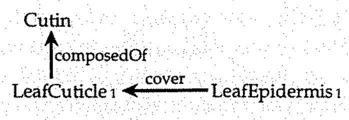
KI interacts with a knowledge engineer to facilitate adding new general statements about the domain to the knowledge base and explore the consequences of the new information. When a knowledge engineer provides new information, KI uses the existing knowledge base to identify possible gaps or conflicts between new and existing knowledge and to identify beliefs supported by the new information. By identifying beliefs supported by the new information KI helps to verify that the actual effect of the new information accords with the knowledge engineer's intended effect. By posing questions back to the knowledge engineer, KI solicits additional information. Thus, KI provides a highly interactive, knowledge-editing interface between the knowledge engineer and the knowledge base that guides knowledge-base development.

KI goes beyond identifying "surface" inconsistencies, such as explicit constraint violations, by determining indirect interactions between new information and existing knowledge. This involves a focused, best-first search exploring the consequences of new information. KI's computational model of knowledge integration comprises three prominent activities:

1. Recognition: identifying the knowledge relevant to new information.
2. Elaboration: applying relevant domain rules to determine the consequences of the new information.
3. Adaptation: modifying the knowledge base to accommodate the elaborated information.

During recognition, KI identifies concepts in the knowledge base that are relevant to the new information. KI uses *views* to determine which concepts, beyond those explicitly referenced by the new information, are relevant (Murray & Porter 1989; Murray 1990; 1995). Each view identifies a set of propositions that interact in some significant way. The views for concepts referenced in the new information determine which existing knowledge structures are recalled during recognition. When new information is presented, KI identifies the views defined for concepts referenced by the new information and heuristically selects one. The concepts contained in the selected view are deemed relevant to the new information, and KI limits its search to consider only the interaction of the new information with the existing knowledge of concepts recalled during recognition.

During elaboration, KI investigates the consequences of the new information for relevant concepts in the knowledge base. This involves applying domain rules defined for the concepts recalled during recognition. Elaboration "expands the information content" of the new information and identifies new views relevant to the elaborated concepts. KI enters a cycle of recognition (i.e., selecting views) and elaboration (i.e., applying domain rules) as it searches for the consequences of new information. Conflicts are revealed when inferences completed during elaboration assert



Numerical subscripts denote class membership (e.g., $isa(LeafEpidermis_1\ LeafEpidermis)$).

Figure 2: The initial learning context

inconsistent conclusions. Novel explanations are detected when the new information enables inferences. Both conflicts and novel explanations suggest learning opportunities.

During adaptation, KI assists the user in modifying the knowledge base to accommodate the elaborated information. In response to conflicts, KI analyzes the support of the conflicting predictions to suggest modifications to the knowledge base that would resolve the conflict. Identifying and correcting conflicts during knowledge integration prevents subsequent problem solving failures. In response to novel explanations, KI evaluates the explanations to suggest ways in which the new information can be generalized or the representations of existing concepts can be augmented.

Through recognition, elaboration, and adaptation, KI determines what existing knowledge is relevant to the new information, the consequences of the new information for the relevant knowledge, and how the relevant knowledge should be modified to accommodate the new information. The following three sections illustrate these activities while describing how KI performs the learning scenario presented in Figure 1. [1]

Recognition

During recognition KI identifies knowledge that is relevant to the new information. This involves maintaining a *learning context* comprising only propositions on concepts deemed relevant to the new information.

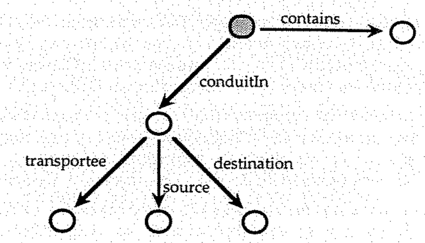
The new information presented to KI is:

$$[\forall\ (x)\ \text{isa}(x\ \text{LeafEpidermis}) \Rightarrow \exists\ (y)\ \text{isa}(y\ \text{LeafCuticle})\ \&\ \text{cover}(x\ y)\ \&\ \text{composedOf}(y\ \text{Cutin})]$$

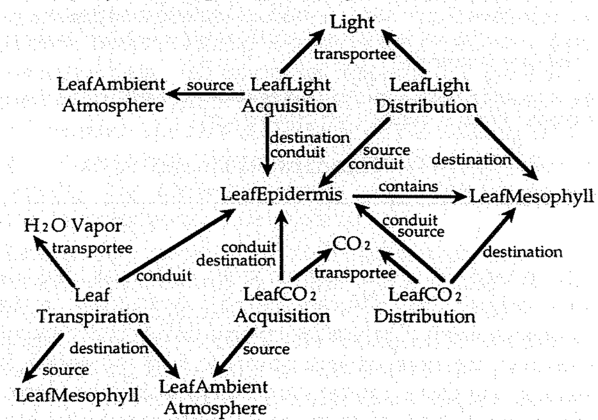
KI initializes the learning context by creating a set of propositions that satisfy the new information (see Figure 2). This involves binding each variable appearing in the new information to a hypothetical instance of the class of objects over which the variable may range. To extend the learning context, KI uses views to determine which concepts in the knowledge base, beyond those explicitly referenced in the new information, are relevant.

[1]This description is simplified for presentation; for a more precise discussion see (Murray 1995).

(3a) Qua Container



(3a) LeafEpidermis Qua Container



The view type *Qua Container* identifies relations that are relevant when considering a concept as a container. The view *LeafEpidermis Qua Container* a semantic network containing those propositions in the knowledge base that are relevant to LeafEpidermis in its role as a container.

Figure 3: An example view type and view

Views are sets of propositions that interact in some significant way and should therefore be considered together. Each view is created dynamically by applying a generic *view type* to a domain concept. Each view type is a parameterized semantic net, represented as a set of paths emanating from a root node. Applying a view type to a concept involves binding the concept to the root node and instantiating each path.

Figure 3 presents an example view type and view. View type *Qua Container* identifies the knowledge base paths emanating from a concept that access properties relevant to its function as a container. These properties include the contents of the container and the processes that transport items into and out of the container. Applying this view type to leaf epidermis identifies the segment of the knowledge base representing the leaf epidermis in its role as a container. For example, this segment includes propositions representing that leaf transpiration transports water vapor from the leaf mesophyll, contained within the leaf epidermis, to the atmosphere outside the leaf epidermis.

rule 1: *If an object is composed of cutin, then it is impermeable to gases.* [∀ (x) composedOf(x Cutin) ⇒ impermeableTo(x Gas)]

rule 2: *If the covering part of an object is impermeable to a substance, then the object is impermeable to the substance.* [∀ (w x y z) cover(w x) & impermeableTo(x y) & unless(partialCover(w x)) & unless(portal(w z) & ¬cover(z x)) ⇒ impermeableTo(w y)]

rule 3: *If the conduit is impermeable to the transportee, then the transportation event is disabled.* [∀ (v w x y z) conduit(v w) & transportee(v x) & isa(x y) & impermeableTo(w y) & unless(conduit(v z) & ¬impermeableTo(z y)) ⇒ status(v Disabled)]

rule 4: *If resource acquisition is disabled, then resource utilization is also disabled.* [∀ (w x y z) acquiredIn(w x) & utilizedIn(w y) & status(x Disabled) & unless(acquiredIn(w z) & ¬status(z Disabled)) ⇒ status(y Disabled)]

rule 5: *If a living thing's method of attaining nutrients is disabled, then it is starving.* [∀ (w x y z) attainerIn(w x) & attainedIn(y x) & isa(y Sugar) & status(x Disabled) & unless(¬health(w Starving)) ⇒ health(w Starving)]

The operator *unless* permits negation-as-failure: *unless(p)* succeeds when *p* cannot be established.

Figure 4: Example domain rules

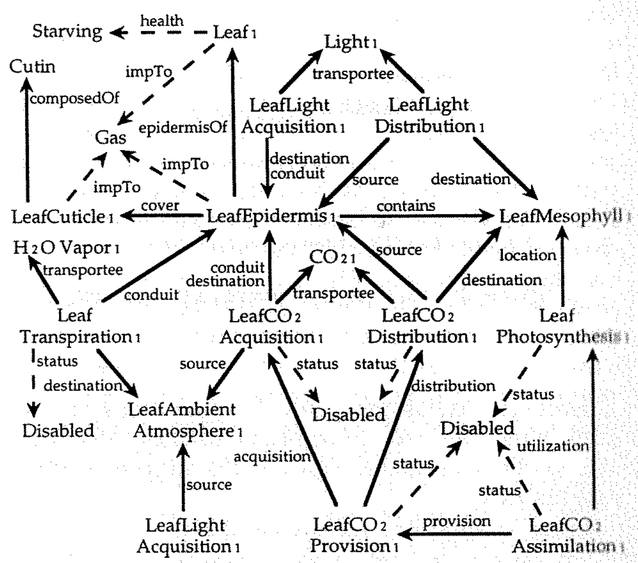


Figure 5: The extended learning context

To extend the learning context, KI identifies the views defined for objects already contained in the learning context. Typically, several different views will be defined for each object, so a method is needed for selecting one from among the many candidate views.

Each candidate view is scored with heuristic measures of its *relevance* to the current learning context and its *interestingness.* Intuitively, relevance is a measure of reminding strength and is computed as a function of the overlap between the candidate view and the context (e.g., the percentage of concepts that are represented in the candidate view that are also represented in the learning context). Interestingness is computed using a small set of heuristics that apply to the individual propositions contained in the view. For example, propositions on concepts referenced by the new information are deemed more interesting than propositions on other concepts. The interestingness of each candidate view is a function of the interestingness of the propositions it contains. The candidate views are ordered by the product of their relevance and interestingness measures, and the view with the highest score is selected. The set of propositions contained in the selected view are added to the learning context. This results in a learning context containing propositions on those concepts in the knowledge base considered most relevant to the new information.

Elaboration

During elaboration KI determines how the new information interacts with the existing knowledge within the learning context. Non-skolemizing rules in the knowledge base are allowed to exhaustively forward-chain, propagating the consequences of the new information throughout the context. For example, one consequence of having a cuticle is that the leaf epidermis is impermeable to gases. Some of the domain rules applicable to this example are listed in Figure 4.

KI enters a cycle of recognition (i.e., selecting views) and elaboration (i.e., applying rules) that explicates the consequences of the new information within an ever-expanding context. The propositions added to the learning context during recognition determine which implicit consequences of the new information will be made explicit during elaboration. This cycle continues until the user intervenes to react to some consequence that KI has identified, or until the computational resources expended exceeds a threshold. Figure 5 illustrates the second round of this cycle. The recognition phase extends the learning context with a view containing propositions that describe how the leaf acquires and makes use of carbon dioxide. The elaboration phase propagates the consequences of the new information throughout the extended context and predicts the leaf cannot perform photosynthesis and starves.

Adaptation

During adaptation, KI appraises the inferences completed during elaboration and assists the user in modifying the knowledge base to accommodate the consequences of the new information. This involves detecting and exploiting learning opportunities that embellish existing knowledge structures or solicit additional knowledge from the knowledge engineer. [2]

A common learning opportunity occurs when inconsistent predictions are made. For example, elaboration reveals that the leaf's cuticle prevents the leaf from acquiring carbon dioxide from the atmosphere. Since

[2]In this example, KI suggests or directly asserts over one hundred new domain rules.

carbon dioxide is an essential resource for photosynthesis, KI concludes that leaves having cuticle cannot perform photosynthesis. This conflicts with the expectation that leaves, in general, must be able to perform photosynthesis. To resolve this conflict, KI inspects the censors of rules participating in the support of the anomalous prediction. Censors are conditions assumed false whenever a rule is used (see the *unless* conditions of rules in Figure 4). Each censor identifies plausible modifications to the knowledge base that would allow the leaf to attain carbon dioxide and perform photosynthesis. Suggesting these plausible modifications prompts the knowledge engineer to provide additional information describing *stomata*, portals in the leaf's epidermis that allow restricted gas flow between the atmosphere and the leaf's interior. This adaptation method is an example of *anomaly-based* abduction: conditions are identified which if assumed to be true would resolve a contradiction.

A second learning opportunity occurs when consequences of the new information suggest generalizations of the new information. For example, elaboration reveals that the leaf cuticle enhances the leaf's physiology by restricting water loss through transpiration. KI recognizes this as a *teleological consequence* of the new information: the physiological benefit of moderating water loss explains why the leaf has a cuticle. A weakest-preconditions analysis of the explanation supporting this conclusion shows that other organs of a plant's shoot system (e.g., stems, fruit, flowers) will also benefit from having a cuticle, and KI suggests this generalization to the knowledge engineer. Consequently, the knowledge structures representing stems, fruit and flowers are embellished to denote they also have cuticles. While this adaptation method constitutes a form of abductive learning, no anomaly is involved. Abduction is motivated by the completion of a teleological explanation rather than a contradiction or a failure to explain some observation.

A third learning opportunity occurs when a property of a particular object in the learning context can be generalized into a property for every instance of a class of objects. For example, elaboration reveals that the hypothetical leaf cuticle is assumed to be translucent. [3] By analyzing the explanation of why the leaf cuticle is translucent, KI determines that all leaf cuticle are, by default, translucent. Consequently, KI asserts the inheritance specification that all instances of leaf cuticle are assumed to be translucent. This is an example of explanation-based learning (EBL) (Mitchell, Keller, & Kedar-Cabelli 1986). However, unlike existing EBL systems, compilation is not triggered when an instance of some specific goal concept has been established. Instead, compilation occurs whenever a sequence of inferences can be summarized by a rule having a specific format. In this case, the permitted format is restricted to the inheritance specification: $(\forall(x)\ isa(x\ Y) \Rightarrow s_1(x\ Z))$, for arbitrary binary predicate s_1, class Y, and constant Z. Speed-up learning (Dietterich 1986) has occurred since a chain of rule-firings which collectively reference an extensive set of preconditions has been compiled into a single rule having an exceedingly simple precondition.

[3] This assumption is made when KI determines that light energy, (typically) used by the leaf during photosynthesis, is acquired from the atmosphere and must pass through the leaf cuticle.

Discussion

This example illustrates how a tool for knowledge integration assists adding a new general rule to a large knowledge base. KI identifies appropriate generalizations of the new information and resolves indirect conflicts between the new information and existing knowledge. It exploits the existing domain knowledge to determine the consequences of new information and guide knowledge-base development.

Intuitively, recognition and elaboration model the learner's *comprehension* of new information. In this model, comprehension is heavily influenced by the learner's existing knowledge: recognition determines what existing knowledge is relevant to the new information; elaboration determines what beliefs are supported by combining the new information with the knowledge selected during recognition. Thus, comprehension produces a set of beliefs that reflect how the new information interacts with existing knowledge. This set of beliefs, and their justifications, afford many diverse learning opportunities, which are identified and exploited during adaptation. This accords with the goals of *multistrategy learning* (e.g., (Michalski 1994)).

The computational model of learning presented in this paper and embodied in KI does not require a priori assumptions about the use of the new information. Existing approaches to compositional modeling (e.g., (Falkenhainer & Forbus 1991; Rickel & Porter 1994; Iwasaki & Levy 1994)) require a given goal query to guide the creation of a domain model, and traditional approaches machine learning (e.g., (Michalski 1986; Dietterich 1986)) exploit assumptions about the eventual uses of acquired knowledge to determine what is to be learned. While such assumptions, when available, can provide substantial guidance to learning, they are not appropriate in many learning situations (e.g., reading a newspaper or textbook). KI relies on *generic learning goals*, such as resolving conflicts (to promote consistency) and embellishing the knowledge of the system's explicitly represented (i.e., named) concepts (to promote completeness). Consequently, this model of learning is not limited to skill refinement, where learning necessarily occurs in the context of problem solving, but can also be applied to learning activities where the eventual uses of acquired knowledge are as yet unknown (Lenat 1977; Morik 1989).

Empirical Analysis

This section presents an empirical analysis of KI's learning behavior during several learning episodes. The learning trials used for this analysis fall into three categories (Murray 1995):

1. The first three trials are *scripted trials.* These trials were deliberately engineered to demonstrate learning behaviors that exemplify learning as knowledge integration. For each, a targeted learning behavior was identified and the knowledge base was extended and corrected as necessary to support that learning behavior.
2. The fourth through the tenth learning trials are *representative trials.* These were developed as a coherent progression of knowledge base extensions thought to be representative for developing a botany knowledge base. For these trials, minor modifications to the knowledge base were performed in order to facilitate reasonable behaviors. This included, for example, correcting pre-existing knowledge-base errors that prevented any reasonable interpretation of the new information and launched the subsequent search for consequences in spurious directions.
3. The eleventh through the seventeenth learning trials are *blind trials.* These were desired knowledge-base extensions submitted by knowledge engineers developing the Botany Knowledge Base. No modifications to the knowledge base were performed to facilitate these trials.

Each group of learning trials has a significantly different origin and extent to which the knowledge base was modified to facilitate desired learning behaviors. Consequently, the following empirical analyses include separate consideration for each of these three groups.

Diversity of learning behaviors

KI was designed to exploit a method of searching for the consequences of new information that was not dedicated to a single adaptation method. The methods for elaboration and recognition reveal the consequences of new and relevant prior knowledge; a suite of adaptation methods then searches these consequences for learning opportunities. This approach separates the search for the consequences of new and prior knowledge from the detection and exploitation of learning opportunities. This separation affords a single, uniform method for identifying consequences that can be used seamlessly and concurrently with a variety of adaptation methods and thus supports a variety of learning behaviors.

To provide evidence for this, the frequencies for each type of learning opportunity that was detected and exploited during the examples are summarized in Figure 6. The data indicate that the learning opportunities were both substantial and diverse: a variety of learning behaviors were exhibited during the learning trials as demonstrated by the diversity of the types of knowledge acquired.

Trials	tax	inh	skol	arg	teleo	abd	total
1 – 3	2.7	9.7	53.0	4.0	6.0	1.0	72.6
4 – 10	2.1	20.0	105.1	14.9	1.3	0.7	143.0
11 – 17	3.4	10.6	71.4	11.9	4.9	0.0	100.6
1 – 17	2.8	14.3	82.1	11.7	3.6	0.5	113.0

The average quantities of acquired rules per learning trial by type. Presented are the numbers of acquired taxonomic rules (tax), inheritance rules (inh), skolemizing rules (skol), argument-typing constraints (arg), rules resulting from teleological learning (teleo), rules resulting from other abductive reasoning (abd), and all acquired rules (total)

Figure 6: Scope of learning opportunities

Measuring learning gain

The obligation of every non-trivial learning system is to acquire knowledge beyond the literal content of new information. *Learning gain* is defined as the amount of acquired knowledge (e.g., measured in terms of the number of beliefs asserted or retracted) not included explicitly in the new information; it provides a natural measure to estimate the effectiveness of a learning program. The *relative learning gain* is defined as the amount of knowledge acquired by one agent (e.g., a learning program) beyond that acquired by another (e.g., a knowledge engineer).

To determine the relative learning gain of KI, professional knowledge engineers were recruited to perform the same seventeen learning trials. These knowledge engineers were quite familiar with the representation language but only marginally familiar with botany and the contents of the knowledge base. However, most of these trials involve only a basic and common knowledge of botany.

For each trial, a knowledge engineer was provided with the new information presented both as a semantic network and as a statement in English. The knowledge engineers were free to make any knowledge-base modifications they felt were appropriate and to inquire about either the domain or the contents of the knowledge base. They were encouraged to follow their normal practices when formalizing and entering knowledge.

The number of axioms produced manually by the knowledge engineers was then compared to the number of axioms produced automatically by KI. Figure 7 presents the results of this experiment. The relative knowledge gain exhibited by KI is significant. Overall, KI derives many times more axioms during these learning trials than was derived manually.

Measuring learning utility

While the data in figures 6 and 7 indicate that KI identifies a diverse and relatively large number of learning opportunities during the learning trials, they do not indicate how useful are the new axioms that result from those opportunities. Traditionally, in machine

Trials	KE	KI	gain
1 − 3	5.0	81.3	76.3
4 − 10	10.1	176.4	166.3
11 − 17	17.4	141.3	123.9
1 − 17	12.2	145.2	132.9

The relative learning gain is computed as the difference between the number of axioms produced by KI and the number of axioms developed manually by a knowledge engineer.

Figure 7: Relative learning gain

Trials	all KE	unique KE	all KI	unique KI
1 − 3	3.6	2.2	4.5	3.8
4 − 10	4.3	2.2	4.9	4.6
11 − 17	4.7	4.0	4.5	3.5
1 − 17	4.5	3.2	4.7	3.8

The subjective utility scores for all axioms produced by the knowledge engineer, axioms produced only by the knowledge engineer, all axioms produced by KI, and axioms produced only by KI.

Figure 8: The utility of acquired axioms

learning, evaluating the utility of acquired knowledge is demonstrated by showing that after learning the system's performance has improved on a set of test queries. This approach is problematic for evaluating KI since by design there is no assumed application task with which to test the system's performance. However, a relative measure of utility can be estimated by subjectively comparing the axioms produced by KI with those produced manually by the knowledge engineers.

For each learning trial, the axioms produced by KI that "correspond" to the axioms produced manually by a knowledge engineer were selected. Two axioms correspond if they are the same or if the predicates match and most of the arguments match (e.g., (*genls GroundWater Water*) and (*genls GroundWater PlantAssimilableWater*) correspond). [4]

Next, for each learning trial, the selected KI axioms were compared to the corresponding axioms developed by the knowledge engineer, and three sets of axioms were defined. The first set includes axioms produced both by KI and the knowledge engineer (i.e., those produced by KI that differed from manually produced axiom only by variable names or by the order of literals). The second set includes axioms produced only by the knowledge engineer. The third set includes axioms produced only by KI. For each trial, the second and third sets were randomly labeled as resulting from *Method 1* and *Method 2*.

Finally, for each trial, a knowledge engineer (other than the knowledge engineer who performed the learning trial) assessed the utility of the axioms that were produced by either KI or the knowledge engineer but not both. For each *Method 1* axiom the evaluator was asked to indicate how much she agreed with the statements *This axiom is useful* and *This axiom is subsumed by axioms of Method 2 and the prior knowledge base.* For each statement, the evaluator scored each *Method 1* axiom with an integer ranging from 1 (denoting strong disagreement with the statement) to 5 denoting (denoting strong agreement with the statement). The evaluator was then asked to perform a similar evaluation of the *Method 2* axioms. The axioms that were produced by both KI and the knowledge engineer were given the scores of 5 both for utility and subsumption.

[4]The knowledge engineers did not produce axioms corresponding to the targeted learning behaviors of the first three trials. Therefore, these engineered learning behaviors were not included in this study.

Trials	all KE axioms	useful KE axioms
1 − 3	3.8	4.4
4 − 10	4.8	5.0
11 − 17	4.4	4.4
1 − 17	4.5	4.6

The subjective estimates of the extent to which axioms produced by a knowledge engineer were subsumed by the axioms produced by KI and the prior knowledge base. Column 2 presents the scores for all manually produced axioms. Column 3 presents the scores for those manually produced axioms deemed useful (i.e., by having a utility score greater than 3).

Figure 9: KI's coverage of KE axioms

Figure 8 presents the average utility score for axioms produced by KI and for axioms produced by the knowledge engineer. The overall utility score for axioms produced only by KI was 0.6 (or about 19%) higher than the scores for axioms produced only by the knowledge engineer. This difference is statistically significant at .95 level of confidence.

Figure 9 presents the extent to which axioms produced by the human knowledge engineer were subsumed by axioms produced by KI. In almost every learning trial, both KI and the knowledge engineer produced axioms that transcend the explicit content of the new information. Learning systems that exploit significant bodies of background knowledge are inherently idiosyncratic, and it would be unreasonable to expect that any learning system (e.g., KI) to completely subsume the learning behavior of another learning system (e.g., a knowledge engineer). However, the data indicate that KI was fairly effective at producing axioms during these learning trials that subsume the useful axioms produced by human knowledge engineers. Overall, KI scored a 4.6 out of a possible 5.0 for subsuming the useful axioms produced manually by professional knowledge engineers on these learning trials. Statistical analysis determined that with a 95% confidence coefficient this score would range between 4.4 and 4.8.

Summary

Knowledge integration is the task of incorporating new information into an existing body of knowledge. This involves determining how new and existing knowledge interact. Knowledge integration differs significantly from traditional machine learning tasks because no specific performance task is assumed. Consequently, the learning system must assess the significance of new information to determine how existing knowledge should be modified to accommodate it.

KI is a machine learning program that performs knowledge integration. It emphasizes the significant role of existing knowledge during learning, and it has been designed to facilitate learning from general statements rather than only from ground observations. When presented with new rules, KI creates a learning context comprising propositions on hypothetical instances that model the new information. By introducing additional propositions that model existing knowledge into the learning context and allowing applicable domain rules to exhaustively forward chain, KI determines how the new information interacts with existing knowledge.

Through actively investigating the interaction of new information with existing knowledge KI is capable of detecting and exploiting a variety of diverse learning opportunities. First, KI identifies plausible generalizations of new information. For example, when told that leaves have cuticle, KI recognizes that the cuticle inhibits water loss and suggests that other organs of the plant's shoot system, such as fruit and stems, also have cuticle. Second, KI identifies indirect inconsistencies introduced by the new information and suggests plausible modifications to the knowledge base that resolve the inconsistencies. For example, KI predicts the leaf cuticle will inhibit the intake of carbon dioxide from the atmosphere and disrupt photosynthesis; consequently, KI suggests the leaf epidermis also has portals to allow the passage of carbon dioxide. Third, KI compiles inferences to embellish the representations of concepts. For example, KI suggests every leaf cuticle is translucent since they must transmit the light assimilated into the leaf from the atmosphere and used during photosynthesis. By identifying the consequences of new information, KI provides a highly reactive, knowledge-editing interface that exploits the existing knowledge base to guide its further development.

Acknowledgments

I am grateful to Bill Jarrold and Kathy Mitchell, for their assistance with the empirical evaluation of KI, and to Bruce Porter, who supervised this research and contributed to both its conception and development. This research was performed while the author was enrolled at the University of Texas at Austin and was supported, in part, by the Air Force Human Resources Laboratory under RICIS contract ET.14.

References

Dietterich, T. 1986. Learning at the knowledge level *Machine Learning* 1(3):287–315.

Falkenhainer, B., and Forbus, K. 1991. Compositional modeling: Finding the right model for the job. *Artificial Intelligence* 51:95–143.

Iwasaki, Y., and Levy, A. 1994. Automated model selection for simulation. In *Proceedings of the National Conference on Artificial Intelligence*, 1183–1190.

Lenat, D., and Guha, R. 1990. ***Building Large Knowledge-Based Systems.*** Reading, MA:Addison-Wesley.

Lenat, D. 1977. The ubiquity of discovery. *Artificial Intelligence* 9:257–285.

Michalski, R. 1986. Understanding the nature of learning: Issues and research directions. In Michalski, R.; Carbonell, J.; and Mitchell, T., eds., *Machine Learning: An Artificial Intelligence Approach*, volume II. Los Altos, CA: Morgan Kaufmann Publishers, Inc. 3–25.

Michalski, R. 1994. Inferential theory of learning: Developing foundations for multistrategy learning. In Michalski, R., and Tecuci, G., eds., *Machine Learning: An Multistrategy Approach*, volume IV. Los Altos, CA: Morgan Kaufmann Publishers, Inc. 3–61.

Mitchell, T.; Keller, R.; and Kedar-Cabelli, S. 1986. Explanation-based generalization: A unifying view. *Machine Learning* 1(1):47–80.

Morik, K. 1989. Sloppy modeling. In Morik, K., ed., *Knowledge Representation and Organization in Machine Learning.* Springer-Verlag. 107–134.

Mostow, J., and Swartout, W. 1986. Towards explicit integration of knowledge in expert systems: An analysis of Mycin's therapy selection algorithm. In *Proceedings of the National Conference on Artificial Intelligence*, 928–935.

Murray, K., and Porter, B. 1989. Controlling search for the consequences of new information during knowledge integration. In *Proceedings of the Sixth International Workshop on Machine Learning*, 290–295.

Murray, K. 1990. Improving explanatory competence. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*, 309–316.

Murray, K. 1995. Learning as knowledge integration. Technical Report TR-95-41, Department of Computer Sciences, University of Texas at Austin.

Porter, B.; Lester, J.; Murray, K.; Pittman, K.; Souther, A.; Acker, L.; and Jones, T. 1988. AI research in the context of a multifunctional knowledge base. Technical Report AI88-88, Department of Computer Sciences, University of Texas at Austin.

Rickel, J., and Porter, B. 1994. Automated modeling for answering prediction questions: Selecting the time scale and system boundary. In *Proceedings of the National Conference on Artificial Intelligence.*