Steps Towards a 2nd Generation Learning by Reading System

Kenneth D. Forbus, Kate Lockwood, Abhishek Sharma, Emmett Tomai

Qualitative Reasoning Group
Northwestern University
2133 Sheridan Road, Evanston, IL, 60208, USA
forbus@northwestern.edu; kate@u.northwestern.edu; a-sharma@northwestern.edu; etomai@u.northwestern.edu

Abstract

Learning by reading is an important scientific problem because it requires modeling a wide range of human abilities. It also could break the knowledge engineering bottleneck, enabling the bootstrapping of intelligent systems via interaction with people using natural language. This paper outlines our progress on creating a 2nd generation learning by reading system, focusing on three main areas: *Multimodal knowledge capture*, *reasoning for NL understanding and learning*, and *analogical dialogue acts*.

Introduction

Learning by reading is an important scientific problem because it requires capturing a wide range of human abilities. It also provides a means of breaking the knowledge engineering bottleneck, potentially enabling the bootstrapping of intelligent systems via interaction with people using natural language. For creating systems that can be treated as long-term collaborators, where the system adapts to its humans rather than the other way around, the generality and flexibility of natural language interaction seems essential. Consequently, there has been growing interest in research on learning by reading.

One fundamental distinction in work so far has been the breadth of materials/depth of understanding tradeoff. On one extreme are the broad/shallow systems, i.e., systems which use a broad range of materials (i.e., slices of the web) and tried to extract only very simple facts from them. Examples include KnowItAll [Etzionoi *et al* 2005] and Factivore [Matuszek *et al* 2005]. For example, KnowItAll processes large numbers of web pages, extracting recurring word triples. These triples can be inspected and used in some ways that object/ attribute/value triples can be. Factivore identifies knowledge gaps by comparing existing instances of a concept (e.g., politicians), and uses NL generation to construct search engine queries to find sentences that can potentially provide missing facts. A refinement process involving combining multiple sources and a human-in-the-loop vetting process ensure quality.

On the other extreme are the narrow/deep systems, i.e., systems which handle only a narrow range of materials (either in terms of simplified NL syntax or being limited to a single domain), but try to extract as much knowledge as possible from those materials. Examples include Learning Reader [Forbus *et al* 2007] and Mobius [Barker *et al* 2007]. Learning Reader is reviewed below. Mobius was a pilot experiment, examining what could be extracted from short paragraphs of unrestricted text, limited to the topic of how hearts work.

The work described here lies on the narrow/deep end of this space, based on our experience in building (with others) Learning Reader. While additional experiments with Learning Reader are in progress, what we have learned so far has led us to begin the creation of a second generation learning by reading system. This paper describes our progress to date on that effort. We begin by briefly reviewing Learning Reader and its strengths and weaknesses. We then outline the design of the *Explanation Agent* (EA), our 2nd-generation system currently under construction. We then describe three areas of work in progress: *multimodal knowledge capture*, *reasoning for NL understanding*, and *analogical dialogue acts*. We close with a discussion of future work.

Learning Reader: A 1st-Generation LbR system

Learning Reader takes as input short stories, written in simplified English. It extracts knowledge from them, using a direct memory access parser (DMAP) [Livingston & Riesbeck, 2007]. DMAP exploits a massive set of phrasal patterns linking KB concepts to how they are expressed in language, automatically extracted from NL knowledge in ResearchCyc. The system's understanding of what it learned was tested by using parameterized questions.

A unique feature of Learning Reader is a model of *rumination*, i.e., the process of assimilating learned knowledge by asking itself questions. Questions arise from knowledge patterns (e.g., in learning world history, the standard journalists' questions – *who*, *what*, *when*, *where*, *why* – are reasonable things to ask about every event) and from comparison with prior experience.

Efficient inference is a key problem in LbR systems. Most practical AI systems are made efficient through careful hand-crafting of representations and hand-tuning of reasoning procedures. This is not an option for LbR systems, since having a human in the loop at this level of detail simply does not scale. Consequently, we developed

Forbus, K., Lockwood, K. and Sharma, A. (2009). Steps towards a 2nd generation learning by reading system. *AAAI Spring Symposium on Learning by Reading*, Spring 2009.

techniques for automatically extracting sets of axioms from knowledge bases for reasoning, based on existing KB contents and what kinds of facts might be gleaned via future reading [Sharma & Forbus, in preparation].

As described in [Forbus *et al* 2007], in one experiment Learning Reader was given a corpus of 62 stories (956 sentences) about the geography, history, and current events in the Middle East. Based on the structure of the parameterized questions and the entities mentioned in the stories, a quiz of 871 questions was automatically generated. Before reading, Learning Reader was able to answer 10% of the questions with 100% accuracy, based on its initial knowledge base contents. After reading, it answered 37% of the questions, with 99.7% accuracy. After reading plus deductive rumination, it answered 50% of the questions, with 99.3% accuracy. Adding in a non-deductive form of rumination (essentially accepting all non-falsifiable analogical inferences as true) gained another 10%, but at the cost of dropping accuracy to 90.8%, since half of the newly-derived answers were incorrect. This experiment shows that Learning Reader is clearly capable of learning from reading, and that rumination can improve understanding in an LbR system.

Analysis of strengths and weaknesses

Learning Reader showed several important things. (1) It showed that deep-understanding learning by reading is possible for medium-sized corpora of simplified English texts. (2) It showed that the contents of ResearchCyc can be useful in natural language experiments. We use our own reasoning engine instead of Cycorp's, since ours is optimized for our purposes, but our knowledge base is created from knowledge extracted from ResearchCyc plus our own extensions for analogical and qualitative reasoning. Without ResearchCyc, Learning Reader would not have been possible. (3) It showed that the DMAP approach can scale to large knowledge bases. Previous attempts at using DMAP were on hand-constructed frame-systems, the largest being on the order of 10^3 axioms. Being able to operate with ResearchCyc contents represents a factor of 1,000 scale-up. (4) Learning Reader's model of rumination seems likely to be applicable to any LbR system. (5) It demonstrated the automatic extraction and optimization of axioms, given patterns of queries to be handled and information about what patterns of facts might be learned.

Learning Reader also had several weaknesses, which illustrate gaps in the scientific understanding and the engineering needed for LbR systems. Currently DMAP system is good at handling simple facts and stories about events, but has trouble handling quotation and complex nested relational patterns[1]. Parameterized questions provided a useful batch evaluation mechanism, but are too restrictive for the range of questions one would like to use for evaluating a system's knowledge. For example, using analogical retrieval and generalization in rumination sometimes produced questions that, if answered, would have provided important insights for the system, but such questions fell outside the range of the parameterized questions. Rumination was purely local, looking only at what could be filled in about each new story. Detecting that a misunderstanding has occurred by comparing problems across a set of stories is beyond its capabilities, as is debugging such misunderstandings. More basically, Learning Reader was not constructed to handle many features common in explanatory texts, such as diagrams and analogies. While experiments with Learning Reader continue, we are also working on a second-generation LbR system, described next.

Explanation Agent: A 2nd-Generation LbR system in progress

The Explanation Agent has its roots in experiments aimed at understanding the roles of qualitative representation in natural language semantics. The EA NLU system (Kuehne & Forbus, 2004) uses Allen's parser [Allen, 1994] and ResearchCyc KB contents. We developed a simplified English dialect, QRG-CE (QRG Controlled English), to factor out complex syntactic structures so that we could better focus on semantics. We showed that QP theory provided a formal semantics for a number of English constructions, in ways that are broadly compatible with linguistic accounts such as FrameNet, and that QP theory descriptions could be automatically constructed via NLU using QRG-CE. This success with simplified language inspired its subsequent use in Learning Reader.

The development on EA NLU continued, motivated both by exploring qualitative models in semantics and to reduce tailorability in stimuli for cognitive simulations. For example, to handle complex scenarios (e.g., used in psychological studies of moral decision-making [Dehghani *et al* 2008]), EA NLU was extended with ideas from Discourse Representation Theory [Kamp & Reyle, 1993], enabling it to handle tense, quotation, counterfactuals, and complex relational structures. For example, the sentence "Because of a dam on a river, 20 species of fish will be extinct." is understood with the appropriate nested quantifiers and the introduction of an intentionally specified set of species, each member of which participates in an extinction event. This level of understanding is much closer to what is needed for science and engineering textbooks and manuals.

We have also been developing the *Companions* cognitive architecture (Forbus & Hinrichs, 2006; Forbus, Klenk, & Hinrichs, 2008) to explore the roles of analogy and qualitative reasoning in cognitive architecture. In Companions, our goal is to create "software organisms", i.e., systems that interact and learn with people over extended periods of time, operating more like collaborators than like tools. Companions use coarse-grained parallelism, with agents that perform broad functional roles. Figure 1 illustrates the architecture that has already

[1] Work in progress by Livingston and Riesbeck is tackling this problem.

been used in a number of experiments (Klenk *et al* 2005; Klenk & Forbus, 2007). The Session Manager, which provides facilities for interacting with a Companion, runs on the user's machine, while the other agents run on a cluster. The Facilitator starts up agents and brokers connections between them. The Executive controls the rest of the agents, using an HTN planner/execution system to ascertain what it should be doing and do it. The Session Reasoner works on whatever external problem is being tackled currently. Access to prior experience is handled via a Tickler, an agent which uses the MAC/FAC model of analogical retrieval (Forbus *et al* 1994) to constantly provide the most relevant remindings given the contents of the Session Reasoner's working memory.

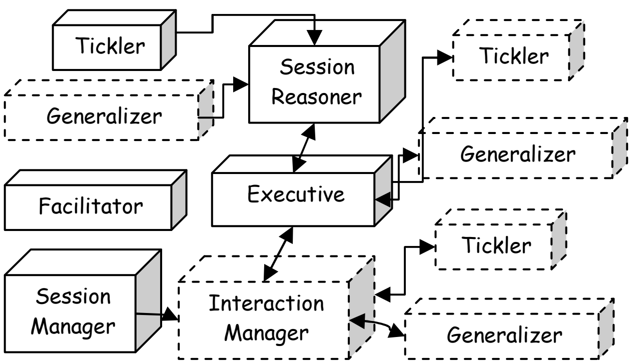

Figure 1: The Companions Architecture (dashed lines show components in progress)

We are currently extending this architecture in several ways, which can be most easily described in terms of new agents. The Interaction Manager handles interaction with the user. We have embedded the EA NLU system into this agent, and are developing new semantic interpretation and dialogue management strategies that exploit the context provided by the architecture. Generalization agents automatically construct generalizations from experience, using SEQL (Kuehne *et al* 2000; Halstead & Forbus, 2005), which produces probabilistic relational representations without the need for pre-defined schema. Ticklers will be added to the Executive and Interaction Manager as well as the Session Reasoner, so that they can exploit prior cases. Generalizers will be added to the Session Reasoner, Interaction Manager, and Executive, to facilitate Companions learning about their domain, interactions with users, and themselves.

This new expansion of the Companions architecture is the foundation for our 2nd generation LbR system, the Explanation Agent. Our focus in the Explanation Agent (hereafter, EA) is understanding complex explanations, such as those found in science texts. Rumination will be implemented as something that the whole system can do when it is not interacting with a user. The rest of this paper describes our efforts in progress on creating this system.

Multimodal Knowledge Capture

Most of the sources that we read contain more than just text – pictures, charts and diagrams all contain important information that contributes to the knowledge that we gain from reading. Studies (e.g. Hegarty & Just, 1993; Mayer & Gallini, 1990) show that people often learn more when presented with a combination of text and diagrams than they do from either modality alone. This suggests that LbR systems should also be able to exploit multiple modalities in texts. We are particularly interested in combinations of text and diagrams as commonly found in textbooks. Consider Figure 2, taken from *Basic Machines* (1994), a physics training manual from the Navy:

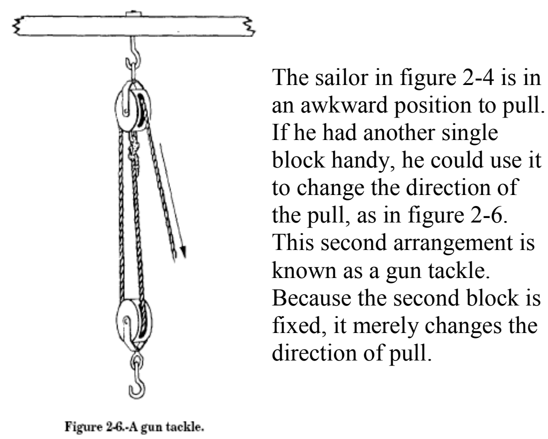

Figure 2: A diagram illustrating a basic machine

The physical arrangement of block and tackle that makes up the gun tackle could have been described in text, but the diagram simplifies communication. Here the physical arrangement shown in the diagram needs to be combined with the function of each individual block, described in the text, to form a comprehensive understanding of the system.

An important property of textbook diagrams is that they often contain a variety of real-world objects, which can be novel or put together in a novel fashion. Labels in the diagram and/or captions aid in understanding. These textual labels, along with other clues from the text, provide scaffolding for the reader on how to integrate information across modalities.

We are currently developing a model of combining information across modalities during knowledge capture. In addition to using EA NLU, we are using the CogSketch sketch understanding system to input all of the diagrams that appear in the source material. Currently, worked solutions and equations from the source text are being hand-represented and added to the system, although in the future, these will be automatically processed as well.

CogSketch

CogSketch (Forbus *et al* 2008) is an open-domain sketch understanding system. Each drawn item in CogSketch is a *glyph*. Glyphs have *ink, content*, and a *name.* Ink consists of the polylines drawn by the user. The content is a token used to represent what the glyph denotes. In CogSketch, users indicate the type of the content of the glyph in terms of concepts from the KB. The name of the glyph is a natural-language string used to identify the glyph linguistically. CogSketch automatically computes a variety of qualitative spatial relationships between the glyphs in a sketch. CogSketch also has specialized glyphs called *annotations* that can be used to attach a numeric dimension to other glyphs in the sketch. For example, Figure 3 shows diagram from *Basic Machines* and its sketched counterpart:

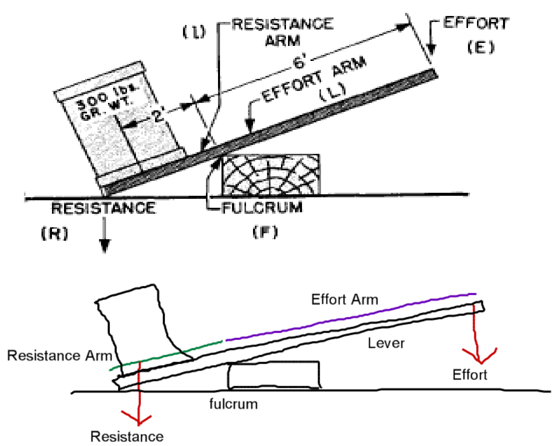

Figure 3 Annotations enable input of forces and numerical dimensions.

In Figure 3, the resistance arm and effort arm are both annotations of the lever glyph. Each has a numerical value and units associated with them, much like the numerical labels in the original diagram.

Experiments and Materials

Our experimental corpus consists of chapters from three instructional texts: Basic Machines [US Navy, 1994], a book on solar energy [Buckley 1979], and an elementary school textbook on optics and light. The texts are being translated to QRG-CE, and diagrams are being drawn using CogSketch. Both Basic Machines and the elementary textbook come with assignments (multiple choice questions) which will be used to test our system. Many of the assignments also include diagrams that must be understood to correctly answer a given question.

While our experiments are still in progress, we outline some interesting problems we have already encountered.

Integrating across modalities: Our current approach is to use analogical matching (see below) to compare the representations of texts and diagrams. Common names in the text and sketch act as required correspondences. The analogical inferences then become conjectures about how information might be combined cross-modally.

Integrating New Concepts. A key role of textbooks is introducing students to new concepts. New concepts are often grounded by tying them to concepts that the reader is presumed to already have. EA will have to correctly connect new concepts to pre-existing concepts. Consider the following set of passages from *Basic Machines:*

> The simplest machine, and perhaps the one with which you are most familiar, is the lever. A seesaw is a familiar example of a lever …

> The three classes of levers are shown in figure 1-2. The location of the fulcrum (the fixed or pivot point) in relation to the resistance (or weight) and the effort determines the lever class.

The first paragraph introduces the concept of a lever and an example (seesaw) is given to help the reader tie the concept to a physical object that they have experienced. An LbR system might have knowledge that bears on the example from prior reading, which provides an opportunity to build on what it has already learned. Truly new concepts can be detected by lack of denotation for the word, e.g., "lever" and "seesaw"[2]. But if there is already some knowledge about levers, the new knowledge must be tested for consistency against the existing knowledge, and tested to see if it allows any standing questions (generated by rumination) to be resolved. Polysemy can lead to interesting problems. For example, consider this passage from Basic Machines:

> The wedge is a special application of the inclined plane. You have probably used wedges. Abe Lincoln used a wedge to help him split logs into rails for fences.

Unfortunately, our knowledge base had only two denotations for the word "wedge": `Wedge-GolfClub` and `SubmarineSandwich`. Clearly, EA should not treat either of these types of wedge as a special application of an inclined plane! Consistency checking needs to detect such cases and trigger the introduction of a new concept (and denotation).

Organizing Information. Information in textbooks is organized into chapters, sections, etc. Clues like headings, indentation, and call-out boxes give human readers cues as to how to organize the information they extract. We plan to exploit this same structure to similarly help EA.

Disambiguation. A benefit of multiple modalities is that each can help disambiguate the other. Unfortunately, sometimes even with both the text and the diagram, there will be ambiguity. Consider again the diagram and text in Figure 2. There are two blocks (pulleys) in the problem. There are two possible ways to distinguish between them.

[2] Not all of the concepts in the ResearchCyc KB are tied into the NL lexicon, so there could be a preexisting concept without the denotation. Looking for such overlaps should be part of the job of rumination, we think.

Since the text refers to adding a pulley to a previous setup, EA could refer back to the previous diagram and realize that the "second block" is the one that is in this diagram, but not the other. Or, EA could use the phrase "Because the second block is fixed …" along with basic knowledge of what it means for an object to be fixed. Both of these strategies, which are natural for people, are needed in LbR systems.

Reasoning for NL understanding and for Learning

EA must be able to reason with its knowledge, which we will test by asking questions, including the ability to ask follow-up questions. It must also identify gaps in its knowledge when it fails to answer questions, and formulate learning goals to drive subsequent reading and rumination. This section describes progress on these issues.

Reasoning for NL Understanding

Syntactic natural language patterns are often ambiguous. Prepositions like 'in' and 'to' can be used to mean different things depending on the context. EA NLU, like many other NLU systems, uses generic, underspecified relationships to provide a minimal interpretation. Consider for example this sentence and its interpretation:

Text: The heat flows to the cylinder.

```
(to-UnderspecifiedLocation flow9172
         cylinder9275)
(isa flow9172 FluidFlow-Translation)
(isa cylinder9275 Cylinder)
```

Underspecified predicates avoid inappropriate commitments during parsing, but are problematic for reasoning. Most axioms used for deductive reasoning require more concrete predicates. Underspecified predicates can also make analogical matching less accurate, since they lead to alignments between entities that are unlikely to be productive. Consequently, we need techniques for what Cycorp calls *predicate strengthening*. Using co-occurrence statistics involving predicates and collections, we have developed a method for estimating the most plausible specialization of a predicate. For the previous example, our algorithm can change the first assertion to `(toLocation flow9172 cylinder9275)`. This method can also identify missing knowledge or imperfect understanding. For example, consider the following sentence and a part of its interpretation.

Text: Radiation is the method by which heat escapes from the planet to space.

```
(by-Underspecified method3668 heat3718)
(isa method3668 TechniqueType)
(isa heat3718 ThermalEnergy)
```

Our method predicts that none of the specializations of `by-Underspecified` are expected to contain instances of `TechniqueType` and `ThermalEnergy`. Therefore, we can conclude that either the relevant predicate is missing or the interpretation is erroneous. How well this algorithm scales is an open question at this point, and it seems likely that backtracking must be supported. It is clear that such a method should also analyze the predicates and collections in the interpretation and estimate the likelihood that they could co-occur.

Reasoning for Learning

AI systems should be cognizant of gaps in their understanding. Missing assertions can often be identified by searching for disconnected components in the interpretations. In such cases, the aim is to infer an assertion about the disconnected entity and other entities in the discourse. We have found that this process is aided by two kinds of heuristic knowledge:

Constraints on event structure: Event structure plays an important role in identifying the missing knowledge. The ResearchCyc KB provides useful information about the expectations of predicates. For example, given an instance of `Translocation`, we should expect predicates like `fromLocation` and `toLocation` about that event. The task of the inference algorithm is to identify which entities in the discourse best fit the arguments of expected predicates. Argument constraints of predicates are not sufficient to narrow down the choices. We have designed a method for analyzing the ground facts to identify the most suitable argument for the predicate. This method relies on co-occurrence of collections and predicates in axioms and ground facts. For example, we can infer that instances of collections like `AscendingStairs` and `Aspirating` are less likely to appear as the first argument of `fromLocation` than instances of collections like `CrossingABorder` and `Smuggling`. This method depends upon bottom-up propagation of evidence from ground facts and axioms.

Commonsense knowledge: Knowledge about the physical structure of the world also helps in identifying suitable options for the arguments of expected predicates. For example, given an instance of `Runway` and an `Airport-Physical`, we could assert that the runway is a physical part of the airport[3]. Similarly, for an event of type `SuicideAttack`, it should be possible to identify at least one patient of the event. Moreover, we can use different kind of knowledge to rule out some alternatives. For example, an instance of `SingleDoerAction` cannot have more than one agent. This increases the likelihood that other entities could be the patients of such events. Similarly, it is not possible to have both `(eventOccursAt x y)` and `(eventOccursAt x z)` when `y` and `z` are disjoint.

Learning new facts which could extend the deductive closure is a very interesting problem. Traditional inductive learning methods need many examples for constructing a model of the target concept. In an open-ended question answering system, it is difficult to come up with the small

[3] The KB has information which helps us to reason that runways are physical parts of airports.

set of target concepts because the list of potential queries is unknown. Therefore, we believe that a different method for learning to reason is needed.

We are using a connection graph approach to suggest plausible deductive rules. In our approach, we construct a connection graph from ground facts and use a spreading activation algorithm to find paths. For example, `(biologicalMother ChelseaClinton HillaryClinton)` will be represented by two nodes `ChelseaClinton` and `HillaryClinton`. The edge between these nodes is labeled `biologicalMother`. A query like `(acquaintedWith BillClinton HillaryClinton)` would involve representing the KB as a graph and finding a path between the nodes labeled `BillClinton` and `HillaryClinton`.

Of course, spreading activation networks do not have any mechanism for enforcing consistency on possible inferences. For example, path 1 shown below is relevant for our query. However, path 2 is very general and cannot be used for inferring any specific information.

```
(biologicalMother ChelseaClinton HillaryClinton),
(father ChelseaClinton BillClinton)     …..(Path 1)

(familyName HillaryClinton "Clinton"),
(familyName BillClinton "Clinton")       …(Path 2)
```

Therefore, we need to augment the expressive power of connection graph methods by a filtering mechanism to prune incorrect inferences. In this example, we need to infer that predicates like `biologicalMother` and `father` are relevant for inferring the predicate in the target query `acquaintedWith` whereas predicates like `familyName` are not.

We believe that knowledge patterns written in terms of high-order predicates can play an important role in enforcing consistency. Such patterns, if extracted from existing axioms, would help in extending the deductive closure by generalization. For example, the rule below represents that predicates belonging to the collection `FamilyRelationSlot` could be combined to justify predicates of type `PersonalAssociationPredicate`.

```
FamilyRelationSlot* → PersonalAssociationPredicate
```

This rule identifies path 1 as plausible because `biologicalMother` and `father` are instances of `FamilyRelationSlot` and `acquaintedWith` is an instance of `PersonalAssociationPredicate`. On the other hand, the predicate in path 2, `familyName`, does not satisfy the constraints of any rule.

Currently we are working on such a method which uses these high level predicates for enforcing consistency. We suspect that a number of diagnostic strategies will be needed to understand errors and prevent them in future inferences. For example, assertions like `(bordersOn SeaOfJapan KoreanPeninsula)` and `(bordersOn SeaOfJapan Russia)` cannot be used to infer `(bordersOn Russia KoreanPeninsula)`. A LbR system should be able to learn that `bordersOn` is not a transitive relation. Our current focus is on finding a small set of learning strategies which could help us in diagnosing incorrect answers and identifying missing facts.

Analogical Dialogue Acts

Analogies are commonplace in explanatory texts. Distant cross-domain analogies are often used to explain novel ideas, as when water flow is used to explain electricity. Within-domain analogies are also commonly used to connect ideas in useful ways. For example, Figure 4 illustrates how an analogy with mining gold can be used to explain the economics of solar energy. The first paragraph starts with a question, setting up the target domain of the analogy, and then introduces the base domain of the analogy, which presumably is more familiar to the reader. The second paragraph starts laying out the correspondences of the analogy, then introduces several inferences. The third paragraph details further inferences that the author wants you to draw from the analogy. Such analogies are a powerful way to communicate complex ideas. One of our goals is to understand how this process works.

> If sunlight is free, why hasn't solar energy been used before to heat houses and produce electricity? […] Let's suppose you owned a gold mine that contained only very low-grade gold ore. You would have to do a lot of digging before you got even a little gold. If you bought some expensive mining equipment, you could process much more of the low-grade ore and get more gold. Thus, even though the gold itself is free, it would be very costly to get very much of it out of the mine. […] Many gold mines in Utah that have only low-grade ore were closed down years ago; some have recently reopened because the price of gold has increased enough to make mining it worthwhile.
>
> Solar energy is like the low-grade ore. The sun's rays must be "mined," or collected, and then transformed into useful heat or electricity before they are worth anything. A solar energy system helps you get "free" solar energy, just as the mining equipment helps you get "free" gold. But, like mining equipment, a solar energy system can be very expensive - perhaps more than the sun's energy is worth.
>
> […] Just as many gold mines in Utah were shut down when it became too costly to mine their gold, many solar hot-water heaters used in Florida and California during the 1950s were shut down when the cost of electricity was so cheap. Recently the cost of heating by gas, oil, and electricity has risen so much that solar energy systems are once again worthwhile, much as the Utah gold mines have once again become profitable because of the rise in the price of gold.

Figure 4: A complex analogy from [Buckley, 1979]

Our working hypothesis is that there are a set of *analogical dialogue acts* that are used in communicating analogies. Like other dialogue acts, they have criteria by which they can be recognized, and a set of implied commitments and obligations for the dialogue participants. We believe that their organization follows directly from the concepts of Gentner's (1983) structure-mapping theory. In structure-mapping, analogical matching takes as input two structured, relational representations, the base and target, and produces as output one or more mappings. Each mapping consists of a set of *correspondences*, identifying how entities and statements in the base align with entities and statements in the target. Mappings can also contain *candidate inferences*, statements in the base that are projected onto the target, using the correspondences of the mapping. The candidate inferences represent conjectures about the target, and constitute a source of analogy's generative power. Here is the initial catalog of analogical dialogue acts we are working with, focusing on their functional role in communicating the intended analogy:

Introduce base/target: The descriptions to be matched must be introduced. Depending on circumstances, this can be as telegraphic as naming them, describing each in detail up front, or anywhere in between. One of the key problems is that the sender and recipient may have quite different models of the base as well as the target, and can construe either of them (as expressed linguistically) in multiple ways using their knowledge of the world. If the linguistic cues are ambiguous, dynamic case construction (Mostek *et al* 2000) could be used to search for construals of both base and target that lead to significant overlap. For instructional texts, the recipient is assumed to not understand the target well, if the analogy is to be informative, so introducing the base tends to be done more carefully, to circumscribe the intended content and perhaps to fill in some details of the base that the recipient might not easily derive or know (e.g., closing of mines in Utah).

Introduce/block correspondence: These acts describe what is intended to go with what in the analogy, or block correspondences that a listener might otherwise be tempted to make. In Figure 4, "Solar energy is like the low-grade ore" is an example of introducing a correspondence.

Introduce/block candidate inference: These acts describe what inferences the sender intends the recipient to draw, or blocks inferences that otherwise might be tempting. The last sentence of the second paragraph of Figure 4 is an example of candidate inference introduction.

Introduce alignable difference: Alignable differences are differences that are connected an analogy's correspondences (Gentner & Markman, 1994). Alignable differences are sometimes introduced via diagrams (Ferguson & Forbus, 1995).

We are currently working on identifying linguistic signals for recognizing analogical dialogue acts in text and in text/diagram combinations. For example, a sequential pair of sentences (sometimes separated by a semicolon) with parallel syntax, the first about the base and the second about the target, suggests correspondence introduction. The third paragraph of Figure 5 indicates that these can be combined in interesting ways; it strengthens the analogy by combining correspondence introduction with adding a novel fact about the target, which helps strengthen the believability of the analogy.

We are also working on dialogue management strategies that use structure-mapping operations to respond to, and generate, these actions. For example, we suspect that recipients maintain evolving descriptions of the base, target, and best conjectured mapping, to generate expectations about subsequent exchanges. Our model of analogical mapping, the Structure-Mapping Engine (Falkenhainer *et al* 1989; Forbus *et al* 1994), has several features that support this. SME can incrementally extend its mappings as information is added to the base or to the target, which is important for handling introduction of base and target information. SME can also take in constraints on mappings, i.e., the effect of an accepted correspondence introduction act is to add a `requiredCorrespondence` constraint to SME, and blocking a correspondence will lead to adding an `excludedCorrespondence` constraint.

Discussion

We believe that learning by reading is one of the key scientific problems for artificial intelligence. Systems that can communicate fluently via combinations of text, sketches, and analogies, and can manage the extension of their own knowledge and skills, could revolutionize the way people and software interact. The first generation of LbR systems showed that AI learning by reading systems were now possible. The next generation must expand the range of materials that can be handled, and show that these ideas can scale. For the narrow/deep region of the LbR spectrum, the next step in scale consists of textbook chapters, where a set of interrelated ideas are expressed in a coherent fashion across multiple paragraphs. Our Explanation Agent project is exploring these issues, for explanatory material that combines texts with diagrams, and uses analogy as a communication device.

While this is clearly work in progress, we believe it is promising. Our medium-term goal is for EA to both understand multiple chapters of material, and use material from earlier chapters in learning later chapters.

In the longer term, as both the broad/shallow and narrow/deep regions are better understood, it will be time to tackle the broad/deep region of potential LbR systems. This will involve web-scale text processing and visual processing of bitmaps (thus overcoming the confines of syntax and media types that define the narrow region) and deep conceptual-level representations of the content (thus overcoming the confines of the word-level representations that define the shallow region).

Acknowledgements

This research was funded by the Intelligent Systems program of the Office of Naval Research, and by a seedling from DARPA IPTO. We thank Dedre Gentner for useful conversations.

References

Allen, J. F. 1994. *Natural Language Understanding*. (2nd ed). Redwood City, CA.: Benjamin/Cummings

Barker, K., Bhalchandra, A., Chaw, S., Fan, J., Friedland, N., Glass, M., Hobbs, J., Hovy, E., Israel, D. Kim, D., Mulkar, R., Patwardhan, S., Porter, B., Tecuci, D., and Yeh, P. 2007. Learning by reading: A prototype system, performance baseline, and lessons learned. *Proceedings of AAAI07*.

Buckley, S. (1979). *Sun up to sun down.* New York: McGraw-Hill.

Dehghani, M., Tomai, E., Forbus, K., Klenk, M. (2008). An Integrated Reasoning Approach to Moral Decision-Making. *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence (AAAI)*. Chicago, IL.

Etzioni, O. , Cafarella, M. , Downey, D. , Popescu, A. , Shaked, T. , Soderland, S. , Weld, D., and Yates, A. 2005. Unsupervised Named-Entity Extraction from the Web: An Experimental Study. *Artificial Intelligence*.

Falkenhainer, B., Forbus, K., Gentner, D. ``The Structure-Mapping Engine: Algorithm and examples'' *Artificial Intelligence*, 41, 1989, pp 1-63.

Ferguson, R. and Forbus, K. 1995. Understanding illustrations of physical laws by integrating differences in visual and textual representations. *AAAI Fall Symposium on Computational Models For Integrating Language And Vision*.

Forbus, K., Ferguson, R. and Gentner, D. 1994. Incremental structure-mapping. *Proceedings of the Cognitive Science Society*, August.

Forbus, K., Gentner, D. and Law, K. 1995. MAC/FAC: A model of Similarity-based Retrieval. *Cognitive Science*, 19(2), April-June, pp 141-205.

Forbus, K. and Hinrichs, T. 2006. Companion Cognitive Systems: A step towards human-level AI. *AI Magazine* 27(2): Summer.

Forbus, K., Hinrichs, T. and Klenk, K. 2008. Companion Cognitive Systems: Design Goals and some Lessons Learned. To appear in AAAI Fall Symposium on Naturally-inspired intelligent systems.

Forbus, K., Usher, J., Lovett, A., Lockwood, K., & Wetzel, J. (2008). CogSketch: Open-domain sketch understanding for cognitive science research and for education. *Proceedings of the Fifth Eurographics Workshop on Sketch-Based Interfaces and Modeling*. Annecy, France.

Forbus, K., Riesbeck, C., Birnbaum, L., Livingston, K., , A., and Ureel, L. (2007). Integrating Natural Language, Knowledge Representation and Reasoning, and Analogical Processing to Learn by Reading. *Proceedings of AAAI-07: Twenty-Second Conference on Artificial Intelligence*, Vancouver, BC

Forbus, K., Usher, J., Lovett, A., Lockwood, K., & Wetzel, J. (2008). CogSketch: Open-domain sketch understanding for cognitive science research and for education. *Proceedings of the Fifth Eurographics Workshop on Sketch-Based Interfaces and Modeling*. Annecy, France.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7, 155-170.

Gentner, D., & Markman, A. B. (1994). Structural alignment in comparison: No difference without similarity. *Psychological Science*, 5(3), 152-158.

Halstead, D. and Forbus, K. 2005. Transforming between propositions and features: Bridging the gap. *Proceedings of AAAI-05*.

Hegarty, M. & Just, M.A. (1989). Understanding machines from text and diagrams. In H. Mandl & J. Levin (Eds.). *Knowledge acquisition from text and picture.* Amsterdam: North Holland (Elsevier Science Publishers).

Kamp, H. and Reyle, U. 1993. *From Discourse to Logic: Introduction to Model theoretic Semantics of Natural Language*. Kluwer Academic Dordrecht; Boston.

Klenk, K., Forbus, K., Tomai, E., Kim, H., and Kyckelhahn, B. 2005. Solving everyday physical reasoning problems by analogy using sketches. *Proceedings of AAAI-05*.

Klenk, M. and Forbus, K. (2007). Measuring the level of transfer learning by an AP Physics problem-solver. Proceedings of AAAI-07: Twenty-Second Conference on Artificial Intelligence, Vancouver, BC

Kuehne, S., Forbus, K., and Gentner, D. 2000. SEQL: Category learning as incremental abstraction using structure mapping. *Proceedings of CogSci-2000*. Philadelphia, PA.

Kuehne, S.E., and Forbus, K. D. 2002. Qualitative physics as a component in natural language semantics: A progress report. *Proceedings of the 24th annual Conference of the Cognitive Science Society*, August, 2002.

Livingston, K., and Riesbeck, C.K. 2007. Using Episodic Memory in a Memory Based Parser to Assist Machine Reading, *Working notes, AAAI Spring Symposium on Machine Reading*, AAAI Press.

Lockwood, K., Forbus, K., Halstead, D., and Usher, J. 2006. Automatic categorization of spatial prepositions. *Proceedings of CogSci06*.

Lockwood, K., Lovett, A., and Forbus, K. (2008). Automatic Classification of Containment and Support Spatial Relations in English and Dutch. In *Proceedings of Spatial Cognition*. Freiberg, Germany.

Lockwood, K., Lovett, A., Forbus, K., Dehghani, M., and Usher, J. (2008). Automatic Interpretation of Depiction Conventions in Sketched Diagrams. *Proceedings of the Eurographics Workshop on Sketch-Based Interfaces and Modeling*.

Matuszek, C., Witbrock, M., Kahlert, R., Cabral, J., Schneider, D., Shaw, P., and Lenat, D. 2005. Searching for common sense: Populating Cyc from the web. *Proceedings of AAAI05*.

Mayer, R.E., & Gallini, J.K. (1990). When Is an Illustration Worth Ten Thousand Words? *Journal of Educational Psychology*, 82(4), 15-26.

Mostek, T., Forbus, K. and Meverden, C. 2000. Dynamic case creation and expansion for analogical reasoning. *Proceedings of AAAI-2000*.

Naval Education and Training Program Management Support Activity. (1994). *Basic Machines and How they Work.* New York: Dover Publications, Inc.

Sharma, A., Forbus, K. (in preparation) Automatic extraction of efficient axiom sets from large knowledge bases.

US Navy, 1994. *Basic Machines*. NAVEDTRA 14037