VERY Large Knowledge bases - Architecture vs Engineering

James Hendler
University of Maryland

Jaime Carbonell
Carnegie Mellon University

Douglas Lenat
Cycorp

Riichiro Mizoguchi
Osaka University

Paul Rosenbloom
University of Southern California

PANEL TOPIC

In the past decade, AI research has created important technologies. The annual investment in, ant[1] return from, the several thousand existing systems employing AI technology is in the hundreds of millions of dollars. ... One feature these successful programs have in common is that they work in well-defined domains in which the systems[1] information, or knowledge base (KB), is not extremely large. Typically, AI systems produce their answers based on no more than several hundred facts concerning the area of their expertise. Although this is enough for many interesting problems, algorithmic difficulties have prevented the scaling of AI technology to much larger problems which require rapid access to many thousands or even millions of facts. Such very large knowledge bases (VLKBs) are necessary to many applications however, particularly those motivated by the exponential growth of the information resources, and needs, of our society.

Dealing with extremely large amounts *of* information has long been a challenge to some researchers in the field of AI. For many people, the very phrase "artificial intelligence'' conjures up a vision of an intelligent computer which can provide immediate access to vast amounts of information. Such systems, like the HAL 9000 computer from Arthur Clarke's *2001: A space Odyssey* or the super-human android, Lieutenant Commander Data, of the television program *Star Trek: the Next Generation* remain squarely in the realm of science fiction, but they are never far from the hearts of many AI researchers. In fact, many early researchers in the field set out to create such programs. Their failures led to the realization that to provide intelligent help in dealing with large amounts of information, an AI system must itself have access to large amounts of knowledge. AI scientists call this the "knowledge-is-power" hypothesis or, more simply, "the knowledge principle" (Lenat and Feigenbaum, 1990).[1]

The particular topic of this panel is to explore WHERE such very large knowledge bases (VLKBs) are to come from. The panel will focus on comparing the imperatives for collecting knowledge (particularly broad knowledge across a wide swath of domains), as is being done in the CYC project, as opposed to developing architectures that are intended to learn the knowledge or to glean the knowledge bases fromn existing data repositories. Contrasted with Lenat's CYC project will be approaches including focused knowledge engineering (Mizoguchi), integrated architectures such as the SOAR, project (Rosenbloom), learning (Carbonell), and large, hybrid knowledge and data bases (Hendler).

Relevance

There are several reasons why we believe this topic is particularly relevant at the current time:

- The media attention (a/k/a hype) over the CYC project has caused massive speculation about the possibility of creation of VLKBS. This panel will include Doug Lenat, who can talk about the current status of CYC and current plans for its use.
- Despite the fact that CYC has become almost synonymous with VLKB efforts, there are currently many other efforts to build and use large knowledge bases. This panel

'These first two paragraph? are taken, nearly verbatim, from .1. Hendler, "High Performance Artificial Intelligence,[1]' *Science,* 265, Aug 12, 1994, p. 891. They are used here with permission of the author.

will familiarize the audience with some of the other (perhaps less controversial) approaches being taken including large lexicons, hybrid knowledge/data bases, and the scaling of rule-based approaches.

- Information and knowledge technology have recently been the focus of major articles in the context of the American "National Information Infrastructure" (NII, information superhighway) and the new "Global Information Infrastucture" (GII, infobahn). This panel will use the VLKBs to help expose the audience to some of the issues resulting from the scaling and use of AI "in the large."

Currently, VLKBs are being pursued in numerous subfields of AI. Among these areas are the following, represented by the members of this panel:

- The best-known, and most talked about, effort in VLKBs is the CYC project. Dr. Lenat is the principle architect of this project.
- In machine translation and NL projects, large lexicons are becoming both necessary and available. Building (and learning) such lexicons has been a focus of Dr. Carbonell's work.
- In the "cognitive architecture" area, approaches are being explored to scale up to much larger systems. The SOAR, project, represented by Dr. Rosenbloom, is the most advanced of these architectures and has been examining the issues of scaling to very large rule-based systems.
* Work in Japan resulted in the development of the Electronic Dictionary, the largest machine translation dictionary built to date. Current work is attempting to scale this work into a practical and usable "comrnonsense" knowledge base. Dr. Riichiro Mizoguchi is an active participant in the Japanese project.
- The use of high performance computing systems to support artificial intelligence research has been gaining significant interest in recent years. One of the most advanced projects in this area is Dr. Hendler's PARKA system, which uses parallel supercomputers in the support of massive knowledge bases and hybrid know ledge/data bases.

We believe that this panel, therefore, visits several of the largest current projects in the area of building very large knowledge bases. As such, it should provide a broad background on which to discuss the critical question of the panel - "Where will these very large knowledge bases come from."

Position Statements

Jaime Carbonell, Carnegie Mellon University

Which comes first, the knowledge or the architecture in building large-scale AI systems? The question is not a chicken-and-egg conundrum, but a crucial, if unresolved, scientific issue. The extreme positions might be taken on the one hand by CYC believers to whom knowledge is virtually everything, and other hand by statisticians in tasks such as speech recognition where the holy grail is a very limited form of architecture, often a kind of Markoff or Basyan model, plus limitless training data (not to be confused with knowledge, they tell us). More moderate views stress the importance of both architecture and knowledge, where either may be general or task-specific.

My philosophy derives from Machine Learning. The architecture is - or should be the generator of the vast bulk of the knowledge in any very large-scale knowledge-based system. Why? It is far easier to build learning architectures than to build truly massive but useful knowledge bases. SOAR, for instance, can build a million chunks automatically. PRODIGY builds thousand-case libraries from problem-solving experience. And, both systems actually use their large knowledge bases to solve new difficult problems efficiently. Both architectures are capable of building new knowledge bases fully automatically in new domains. In contrast, the hand-crafted VLKB approach, has no such generative capabilities. However, learning architectures address the problem of the utility and organization of the knowledge they aquire, but fail to address the problem of cross-task generality of that knowlege. This remains one of the greatest challenges of the architecture-first approach, and a reason why there is still room for hand-built knowledge bases.

James Hendler, University of Maryland

Too much of the focus in knowledge base development, to date has been on the form of

the data as it is input, without enough concentration on how to get it back out and how it. can be used. So-called knowledge engineering approaches, like CYC', have focused on development of VLKBs with many different retrieval strategies, few of which scale well and all of which take a long time to retrieve complex facts. So-called architectural approaches, including SOAR and Prodigy, have focused on learning of large amounts of data, and using them in fairly specific ways. Unlike our counterparts in the database community, we've never put too much thought into how humans will access the facts in the VLKBS or how AI systems may use it. If inferences take minutes or hours, then memory becomes a major bottleneck. I therefore will argue that architecture is a primary concern, but not the "knowledge" architecture per se, but rather the computational architectures that will allow rapid access to VLKBS.

Douglas Lenat, Cycorp

Everyone else on this panel has clearly articulated their disagreement with my approach to getting VLKBs, which might be caricatured as "build it carefully and slowly, by hand." But I find 1 must strongly disagree with them, in the sense that, my message to them is: "No, I actually agree with all of you!" Namely, the CYC approach is merely to carry on manual knowledge entering so long as it's needed, to prime the pump, as it were, with knowledge so fundamental that it's easier to just tell the machine those things than to have it induce or deduce them. This is the knowledge which is a prerequisite for "real" automated learning, guided by plausible theories rather than dissociated statistics. It is also the knowledge which is a prerequisite for "real" automated understanding of natural language (and for that matter speech and images as well) and which therefore is required in order to break the chicken-and-egg codependency that Carbonnel refers to. And even if we do our job well, Hendler's sort of efficient reasoning machinery will be a must. The real points of disagreement among our group, apparently, are: (1) Bow much knowledge needs to be manually represented, before automated methods can really take off? 1 think it's quite a bit - several person-centuries' worth of effort - and others think it may be drastically less. (2) Is it important to get the architecture "right"? The others think that the answer is yes that it's critically important in fact but I think the answer is no, that we can merely pick one and get started encoding the knowledge, and it will evolve as that KB-building enterprise unfolds. We have invested well over a person-century of time since the CYC project began in 1984, following through on this philosophy, and its architecture has evolved quite dramatically in that time, in directions I neither expected nor welcomed. I'll try to convey some of the feel for those changes, and give some arguments for the two contrarian points of view I've listed above.

Riichiro Mizoguchi, Osaka University

Preparing for the coming advanced information society, Japan is trying to set, up a national project called "Human media" which aims at building a seamless information space. The future information technology has to cope with huge amount of knowledge represented in multimedia in a unified manner and to provide humans with sophisticated support for traveling around in a huge information space. Through this project, we challenge some innovative research topics such as sharing and reuse of multimedia knowledge, ontology design for bridging the gap between computer media and human media and for integration of multimedia information, building very large knowledge bases based on multi-agent systems, etc, In this panel, 1 would like to talk about the philosophy behind the project and the major research plans towards so-called "content-directed AI".

Paul Rosenbloom, University of Southern California

The most effective way for an agent - either human or synthetic to learn large amounts of knowledge has to be for it to make use of whatever information the world provides to it. Whether information is available in the form of theoretical statements, facts, data bases, experiences, guidance, lectures, books, examples, stories, images, or facial expressions, a failure to extract what, lessons the information has to offer will result in slower growth of the internal knowledge base. 'The problem though is that no existing synthetic agents can actually make use of all of these forms of information (or even a significant fraction of them). The pure knowledge-engineering position responds to this problem by reformulating information by hand so as to make it more easily usable by an agent, while the pure learning position responds by developing automatic reformulation mechanisms (also often called learning, understanding or comprehension mechanisms) that allow agents to directly accept a broader range of information. However,

there is no essential reason why purity of either sort should actually be useful, especially given that some knowledge (such as the Laws of Physics) has taken centuries to extract from the raw data, while other information is quite easily extracted from everyday experience. Our experience in working with Soar agents that learn aligns with this mixed view, in that it is still much easier to spoon feed most conceptualizations of the world into such agents than it is for the agents to learn them autonomously (i.e., the Cyc position?); while conversely, it can be easier for the agents to learn the many variations on a conceptual theme through experience than it is to hand code them (i.e., the Prodigy position?). (In addition, between these two extremes, learning from guided experience can sometimes do a reasonable job on both conceptualizations and variations.) Since there tend to be more variations then distinct conceptualizations, the quantity of information learned can far outstrip the quantity hand coded - in fact, in our experience, by factors of thousands - even when most of the crucial information is hand coded. As Carbonell points out in his note, considerable research is still necessary before agents will be able to autonomously acquire information that is very broad (e.g., involving conceptualizations across multiple domains); however, the same is also true for information that is very deep (e.g., non-obvious scientific theories). In addition, as Hendler points out in his note, research is needed on the efficient and effective retrieval of knowledge, as a function of an agent's goals and situation, from very large knowledge bases; although Doorenbos has at least shown in Soar that it is possible to acquire over a million rules, while still allowing their effective and efficient use.