Appears in *AAAI-96* (abstract of invited talk)

Moving Up the Information Food Chain: Deploying Softbots on the World Wide Web

Oren Etzioni
Department of Computer Science and Engineering
University of Washington
Seattle, WA 98195
http://www.cs.washington.edu/homes/etzioni

Abstract

I view the World Wide Web as an *information food chain* (figure 1). The maze of pages and hyperlinks that comprise the Web are at the very bottom of the chain. The WebCrawlers and Alta Vistas of the world are *information herbivores*; they graze on Web pages and regurgitate them as searchable indices. Today, most Web users feed near the bottom of the information food chain, but the time is ripe to move up. Since 1991, we have been building *information carnivores*, which intelligently hunt and feast on herbivores in Unix (Etzioni, Lesh, & Segal 1993), on the Internet (Etzioni & Weld 1994), and on the Web (Doorenbos, Etzioni, & Weld 1996; Selberg & Etzioni 1995; Shakes, Langheinrich, & Etzioni 1996).

Motivation

Today's Web is populated by a panoply of primitive but popular information services. Consider, for example, an information cow such as Alta Vista. Alta Vista requires massive memory resources (to store an index of the Web) and tremendous network bandwidth (to create and continually refresh the index). The cost of these resources is amortized over millions of queries per day. As a result, the CPU cycles devoted to satisfying each individual query are sharply curtailed. There is no *time* for intelligence. Furthermore, each query is independent of the previous one. No attempt is made to customize Alta Vista's responses to a particular individual. The result is homogenized, least-common-denominator service.

In contrast, visionaries such as Alan Kay and Nicholas Negroponte have been advocating agents — personal assistants that act on your behalf in cyberspace. While the notion of agents has been popular for more than a decade, we have yet to build agents that are both widely used *and* intelligent. The Web presents a golden opportunity and an implicit challenge for the AI community. As the old adage goes "If not us, then who? And if not now, when?"

The challenge of deploying web agents will help revitalize AI and forge closer links with other areas of computer science. But be warned, the Web community is hungry, impatient, and skeptical. They expect:

- **Robustness:** a working system, accessible seven days a week, twenty-four hours a day.
- **Speed:** virtually all widely-used Web resources begin transmitting useful (or at least entertaining) information within seconds.
- **Added Value:** any increase in sophistication had better yield a tangible benefit to users.

Is the Web challenge a distraction from our long-term goal of understanding intelligence and building intelligent agents? I believe that the field benefits from a mixture of long-term and short-term goals and from both empirical and theoretical work. Work toward the goal of deploying intelligent agents on the Web is a valuable addition to the current mix for two reasons. First, the Web suggests new problems and new constraints on existing techniques. Second, intelligent Web agents will provide tangible evidence of the power and utility of AI techniques. Next time you encounter AI bashing, wouldn't it be satisfying to counter with a few well-chosen URLs? Personally, I find the Web irresistible. To borrow Herb Simon's phrase, it is today's "Main Chance." Simon describes his move from the "academic backwater" of public administration to AI and cognitive psychology as "gravitating toward the sun" (Simon 1991, pages 113-114). While AI is not an academic backwater, the Web *is* today's sun. Turning towards the sun and responding to the Web challenge, my collaborators and have begun to deploy a species of information carnivores (called *softbots*) on the Web.

Softbots

Softbots (software robots) are intelligent agents that use software tools and services on a person's behalf (see figure 2 for a softbot family tree). Tool use is one of the hallmarks of intelligence. In many cases, softbots rely on the same tools and utilities available to human computer users — tools for sending mail, printing files, and so on. Mobile robots have yet to achieve the physical analog — using vacuum cleaners,

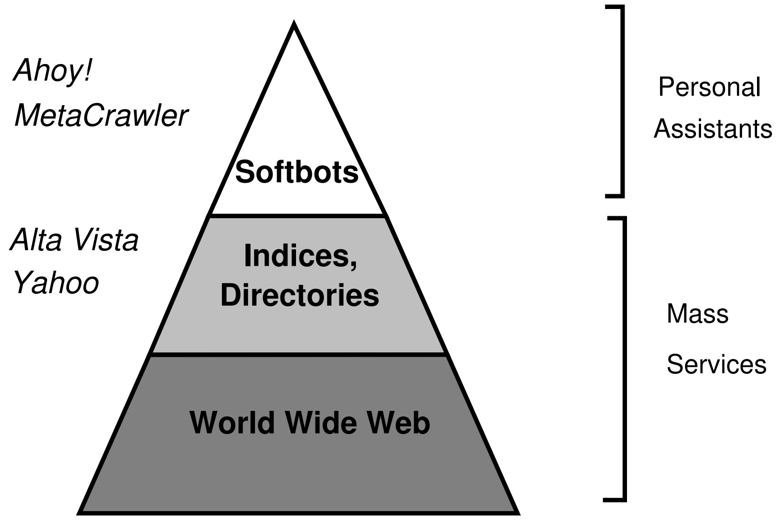

Figure 1: The Information Food Chain

lawn mowers, etc.[1]

Much of our work has focused on the Internet softbot (also known as Rodney) (Etzioni & Weld 1994). Rodney enables a person to state *what* he or she wants accomplished. Rodney disambiguates the request and dynamically determines *how* and *where* to satisfy it, utilizing a wide range of Internet services and Unix commands. Rodney relies on a declarative representation of the different software tools at its disposal, enabling it to chain together multiple tools in response to a user's request. Rodney uses automatic planning technology to dynamically generate the appropriate action sequence. The Internet softbots project has led to a steady stream of technical results (*e.g.*, (Etzioni *et al.* 1992; Etzioni, Golden, & Weld 1994; Golden, Etzioni, & Weld 1994; Kwok & Weld 1996; Perkowitz & Etzioni 1995)). Closely related projects include (Kirk *et al.* 1995; Arens *et al.* 1993).

Unfortunately, we have yet to produce a planner-based softbot that meets the stringent demands of the Web community. While continuing our ambitious long-term project to develop planner-based softbots, we have embraced a new strategy for the creation of intelligent agents which I call "useful first." Instead of starting with grand ideas about intelligence and issuing a promissory note that they will eventually yield useful intelligent agents, we take the opposite tack; we begin with useful softbots deployed on the Web, and issue a promissory note that they will evolve into more intelligent agents. We are still committed to the goal of producing agents that are *both* intelligent and useful. However, I submit that we are more likely to achieve this conjunctive goal if we reverse the traditional subgoal ordering and focus on building useful systems first.

The argument for "useful first" is analogous to the argument made by Rod Brooks (Brooks 1991) and others (Etzioni 1993; Mitchell *et al.* 1990) for building complete agents and testing them in a real world. As Brooks put it, "with a simplified world...it is very easy to accidentally build a submodule of the systems which happens to rely on some of those simplified properties...the disease spreads and the complete system depends in a subtle way on the simplified world." This argument applies equally well to user demands and real-time constraints on Web agents.

There is a huge gulf between an AI prototype and an agent ready for deployment on the Web. One might argue that this gulf is of no interest to AI researchers. However, the demands of the Web community constrain the AI techniques we use, and lead us to new AI problems. We need to recognize that intelligent agents are ninety-nine percent computer science and one percent AI. The AI is critical but we cannot ignore the context into which it is embedded. Patrick Winston has called this the "raisin bread" model of AI. If we want to bake raisin bread, we cannot focus exclusively on the raisins.[2]

Operating on a shoestring budget, we have been able

[1]softbots are an attractive substrate for intelligent-agent research for the following reasons (Etzioni 1993; 1994). First, the cost, effort, and expertise necessary to develop and systematically experiment with software artifacts are relatively low. Second, software environments circumvent many of the thorny but peripheral problems that are inescapable in physical environments. Finally, in contrast to simulated physical worlds, software environments are readily available (sophisticated simulations can take years to perfect), intrinsically interesting, and *real*. However, Softbots are *not* intended to replace robots; Robots and softbots are complimentary.

[2]See (Brachman 1992) for an account of the massive re-engineering necessary to transform an "intelligent first" knowledge representation system into a usable one.

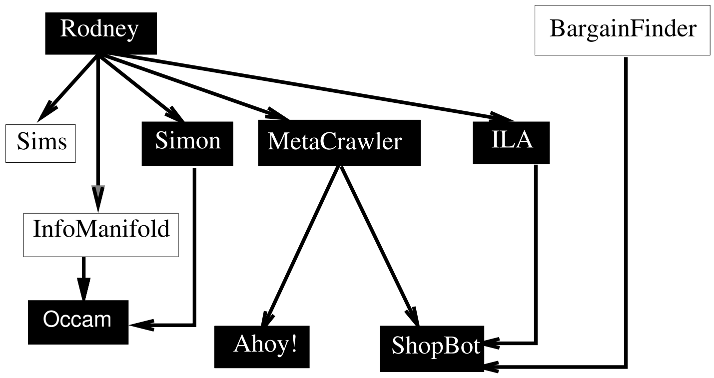

Figure 2: The Softbot Family Tree. The black boxes represent softbots developed at the University of Washington. MetaCrawler, Ahoy!, and ShopBot have been deployed on the Web.

to deploy several softbots on the Web within one year. I review our fielded softbots and then consider both the benefits and pitfalls of the "useful first" approach.

MetaCrawler

The MetaCrawler softbot[3] provides a single, unified interface for Web document searching (Selberg & Etzioni 1995). MetaCrawler supports an expressive query language that allows searching for documents that contain certain phrases and excluding documents containing other phrases. MetaCrawler queries nine of the most popular information herbivores in parallel. Thus, MetaCrawler eliminates the need for users to try and retry queries across different herbivores. Furthermore, users need not remember the address, interface and capabilities of each one. Consider searching for documents containing the phrase "four score and seven years ago." Some herbivores support phrase searching whereas others do not. MetaCrawler frees the user from having to remember such details. If a herbivore supports phrase searching, MetaCrawler automatically invokes this feature. If a herbivore does not support phrase searching, MetaCrawler automatically downloads the pages returned by that herbivore and performs its own phrase search locally.

In a recent article, Forbes Magazine asked Lycos's Michael Maudlin "why aren't the other spiders as smart as MetaCrawler?" Maudlin replied "with our volume I have to turn down the smarts...MetaCrawler will too if it gets much bigger." Maudlin's reply misses an important point: because MetaCrawler relies on information herbivores to do the resource-intensive grazing of the Web, it is sufficiently lightweight to run on an average PC and serve as a personal assistant. Indeed, MetaCrawler-inspired PC applications are now on the market.

[3] `http://www.cs.washington.edu/research/metacrawler`

MetaCrawler demonstrates that Web services and their interfaces may be de-coupled. MetaCrawler is a *meta-interface* with three main benefits. First, the same interface can be used to access *multiple* services simultaneously. Second, since the meta-interface has relatively modest resource requirements it can reside on an individual user's machine, which facilitates customization to that individual. Finally, if a meta-interface resides on the user's machine, there is no need to "turn down the smarts." In a Web-mediated client/server architecture, where intelligence resides in the client, "volume" is no longer a limiting factor on the "smarts" of the overall system.

While MetaCrawler does not currently use AI techniques, it is evolving rapidly. For example, we are investigating the use of document clustering to enable users to rapidly focus on relevant subsets of the references returned by MetaCrawler. In addition, we are investigating mixed-initiative dialog to help users focus their search. Most important, MetaCrawler is an enabling technology for softbots that are perched above it in the information food chain.

Ahoy! **The Home Page Finder**

The Ahoy! softbot[4] specializes in locating people's home pages on the Web by filtering MetaCrawler output (Shakes, Langheinrich, & Etzioni 1996). Ahoy! takes as input a person's name and affiliation, and attempts to find the person's home page. Ahoy! queries MetaCrawler and uses knowledge of Web geography (*e.g.*, the URLs of home pages at the University of Washington end with `washington.edu`) and home page appearance (a home page title is likely to contain a person's last name) to filter MetaCrawler's output. Typically, Ahoy! is able to cut the number of references returned by a factor of forty but still maintain

[4] `http://www.cs.washington.edu/research/ahoy`

very high accuracy.

Since Ahoy!'s filtering algorithm is heuristic, it asks its users to label its answers as correct or not. Ahoy! uses the feedback it receives from its users to continually improve its performance. It rapidly collects a set of home pages and near misses (labeled as such by users) to use as training data for an algorithm that attempts to learn the conventions underlying home page placement. For example, home pages at the University of Washington's Computer Science Department typically have the form `http://www.cs.washington.edu/homes/<lastname>`. After learning, Ahoy! is able to locate home pages of individuals even *before* they are indexed by MetaCrawler's herd of information herbivores.

In the context of Ahoy!, the "useful first" constraint led us to tackle an important impediment to the use of machine learning on the Web. Data is abundant on the Web, but it is unlabeled. Most concept learning techniques require training data labeled as positive (or negative) examples of some concept. Techniques such as uncertainty sampling (Lewis & Gale 1994) reduce the amount of labeled data needed, but do not eliminate the problem. Instead, Ahoy! attempts to harness the Web's interactive nature to solve the labeling problem. Ahoy! relies on its initial power to draw numerous users to it and to solicit their feedback; it then uses this feedback to solve the labeling problem, make generalizations about the Web, and improve its performance. Note that by relying on feedback from *multiple* users, Ahoy! rapidly collects the data it needs to learn; systems that are focused on learning an individual users taste do not have this luxury. Ahoy!'s boot-strapping architecture is not restricted to learning about home pages; user feedback may be harnessed to learn in a variety of Web domains.

ShopBot

ShopBot[5] is a softbot that carries out comparison shopping at Web vendors on a person's behalf (Doorenbos, Etzioni, & Weld 1996). Whereas virtually all previous Web agents rely on hard-coded interfaces to the Web sites they access, ShopBot autonomously *learns* to extract product information from Web vendors given their URL and general information about their product domain (*e.g.*, software). Specifically, ShopBot learns how to query a store's searchable product catalog, learns the format in which product descriptions are presented, and learns to extract product attributes such as price from these descriptions.

ShopBot's learning algorithm is based in part on that of the Internet Learning Agent (ILA) (Perkowitz & Etzioni 1995). ILA learns to extract information from unfamiliar sites by querying with familiar objects and analyzing the relationship of output tokens to the query object. ShopBot borrows this idea from ILA; ShopBot learns by querying stores for information on popular products, and analyzing the stores' responses. However, ShopBot tackles a more ambitious learning problem than ILA because Web vendors are far more complex and varied than the Internet directories that ILA was tested on.

In the software shopping domain, ShopBot has been given the home pages for 12 on-line software vendors. After its learning is complete, ShopBot is able to speedily visit the vendors, extract product information such as availability and price, and summarize the results for the user. In a preliminary user study, ShopBot users were able to shop four times faster (and find better prices!) than users relying only on a Web browser (Doorenbos, Etzioni, & Weld 1996).

Discussion

Every methodology has both benefits and pitfalls; the softbot paradigm is no exception. Perhaps the most important benefit has been the discovery of new research challenges, the imposition of tractability constraints on AI algorithms, and the resulting innovations. In recent years, planner-based softbots have led us to the challenge of incorporating information goals, sensory actions, and closed world reasoning into planners in a tractable manner. Our focus on tractability led us to formulate UWL (Etzioni *et al.* 1992) and Local Closed World Reasoning (Etzioni, Golden, & Weld 1994; 1995). We expect "useful first" to be equally productive over the next few years. For example, MetaCrawler has led us to investigate on-line, real-time document clustering. Previous approaches to document clustering typically assume that the entire document collection is available ahead of time, which permits analysis of the collection and extensive preprocessing. In the context of MetaCrawler, document snippets arrive in batches and the delay due to document clustering has to be minimal. As a result, clustering must take place as the snippets are rolling in.

I acknowledge that our approach has numerous pitfalls. Here are a couple, phrased as questions: will we fail to incorporate substantial intelligence into our softbots? Does the cost of deploying softbots on the Web outweigh the benefit? Our preliminary success in incorporating AI techniques into our deployed softbots makes me optimistic, but time will tell.

Conclusion

Each of the softbots described above uses multiple Web tools or services on a person's behalf. Each softbot enforces a powerful abstraction: a person is able to state *what* they want, the softbot is responsible for deciding *which* Web services to invoke in response and *how* to do so. Each softbot has been deployed on the Web, meeting the requirements of robustness, speed, and added value. Currently, MetaCrawler receives close to 100,000 hits a day. Ahoy! and ShopBot have yet to be announced publicly. However, shortly after its

[5] `http://www.cs.washington.edu/research/shopbot`

release on the Web, Ahoy! was discovered by Yahoo and mentioned in its directory. Immediately, it began receiving hundreds of queries per day.

Having satisfied the "useful first" constraint, our challenge is to make our current softbots more intelligent, inventing new AI techniques and extending familiar ones. We are committed to doing so while keeping our softbots both usable and useful. If we succeed, we will help to rid AI of the stereotype "if it works, it ain't AI." To check on our progress, visit the URLs mentioned earlier. Softbots are standing by...

Acknowledgments

I would like to thank Dan Weld, my close collaborator on many of the softbots described above, for his numerous contributions to the softbots project and its vision; he cannot be held responsible for the polemic tone and subversive methodological ideas of this piece. I would also like to thank my co-softbotists David Christianson, Bob Doorenbos, Marc Friedman, Keith Golden, Nick Kushmerick, Cody Kwok, Neal Lesh, Mark Langheinrich, Sujay Parekh, Mike Perkowitz, Richard Segal, and Jonathan Shakes for making softbots real. Thanks are due to Steve Hanks and other members of the UW AI group for helpful discussions and collaboration. I am indebted to Ema Nemes for her assistance in writing this paper and creating its figures. This research was funded in part by Office of Naval Research grant 92-J-1946, by ARPA / Rome Labs grant F30602-95-1-0024, by a gift from Rockwell International Palo Alto Research, and by National Science Foundation grant IRI-9357772.

References

Arens, Y.; Chee, C. Y.; Hsu, C.-N.; and Knoblock, C. A. 1993. Retrieving and integrating data from multiple information sources. *International Journal on Intelligent and Cooperative Information Systems* 2(2):127–158.

Brachman, R. 1992. "Reducing" CLASSIC to Practice: Knowledge Representation Theory Meets Reality. In *Proc. 3rd Int. Conf. on Principles of Knowledge Representation and Reasoning*.

Brooks, R. 1991. Intelligence without representation. *Artificial Intelligence* 47:139–159.

Doorenbos, B.; Etzioni, O.; and Weld, D. 1996. A scalable comparison-shopping agent for the world-wide web. Technical Report 96-01-03, University of Washington, Department of Computer Science and Engineering. Available via FTP from `pub/ai/` at `ftp.cs.washington.edu`.

Etzioni, O., and Weld, D. 1994. A Softbot-Based Interface to the Internet. *CACM* 37(7):72–76. See `http://www.cs.washington.edu/research/softbots`.

Etzioni, O.; Hanks, S.; Weld, D.; Draper, D.; Lesh, N.; and Williamson, M. 1992. An Approach to Planning with Incomplete Information. In *Proc. 3rd Int. Conf. on Principles of Knowledge Representation and Reasoning*. San Francisco, CA: Morgan Kaufmann. Available via FTP from `pub/ai/` at `ftp.cs.washington.edu`.

Etzioni, O.; Golden, K.; and Weld, D. 1994. Tractable closed-world reasoning with updates. In *Proc. 4th Int. Conf. on Principles of Knowledge Representation and Reasoning*, 178–189. San Francisco, CA: Morgan Kaufmann.

Etzioni, O.; Golden, K.; and Weld, D. 1995. Sound and efficient closed-world reasoning for planning. Technical Report 95-02-02, University of Washington. Available via FTP from `pub/ai/` at `ftp.cs.washington.edu`.

Etzioni, O.; Lesh, N.; and Segal, R. 1993. Building softbots for UNIX (preliminary report). Technical Report 93-09-01, University of Washington. Available via anonymous FTP from `pub/etzioni/softbots/` at `cs.washington.edu`.

Etzioni, O. 1993. Intelligence without robots (a reply to brooks). *AI Magazine* 14(4). Available via anonymous FTP from `pub/etzioni/softbots/` at `cs.washington.edu`.

Etzioni, O. 1994. Etzioni Responds. *AI Magazine*. Response to commentary on "Intelligence without Robots (A Reply to Brooks)".

Golden, K.; Etzioni, O.; and Weld, D. 1994. Omnipotence without omniscience: Sensor management in planning. In *Proc. 12th Nat. Conf. on A.I.*, 1048–1054. Menlo Park, CA: AAAI Press.

Kirk, T.; Levy, A. Y.; Sagiv, Y.; and Srivastava, D. 1995. The information manifold. In *Working Notes of the AAAI Spring Symposium: Information Gathering from Heterogeneous, Distributed Environments*, 85–91. Stanford University: AAAI Press. To order a copy, contact sss@aaai.org.

Kwok, C., and Weld, D. 1996. Planning to gather information. Technical Report 96-01-04, University of Washington, Department of Computer Science and Engineering. Available via FTP from `pub/ai/` at `ftp.cs.washington.edu`.

Lewis, D., and Gale, W. 1994. Training text classifiers by uncertainty sampling. In *17th Annual Int'l ACM SIGIR Conference on Research and Development in Information Retrieval*.

Mitchell, T. M.; Allen, J.; Chalasani, P.; Cheng, J.; Etzioni, O.; Ringuette, M.; and Schlimmer, J. C. 1990. Theo: A framework for self-improving systems. In VanLehn, K., ed., *Architectures for Intelligence*. Hillsdale, NJ.: Erlbaum.

Perkowitz, M., and Etzioni, O. 1995. Category translation: Learning to understand information on the internet. In *Proc. 15th Int. Joint Conf. on A.I.*

Selberg, E., and Etzioni, O. 1995. Multi-Service Search and Comparison Using the MetaCrawler. In *Proc. 4th World Wide Web Conf.*, 195–208. See `http://www.cs.washington.edu/research/metacrawler`.

Shakes, J.; Langheinrich, M.; and Etzioni, O. 1996. Ahoy! the home page finder. Technical report, University of Washington. To appear, see `http://www.cs.washington.edu/research/ahoy`.

Simon, H. 1991. *Models of My Life*. Basic Books.