Beating Common Sense into Interactive Applications

Henry Lieberman, Hugo Liu, Push Singh, Barbara Barry

■ A long-standing dream of artificial intelligence has been to put commonsense knowledge into computers—enabling machines to reason about everyday life. Some projects, such as Cyc, have begun to amass large collections of such knowledge. However, it is widely assumed that the use of common sense in interactive applications will remain impractical for years, until these collections can be considered sufficiently complete and commonsense reasoning sufficiently robust. Recently, at the Massachusetts Institute of Technology's Media Laboratory, we have had some success in applying commonsense knowledge in a number of intelligent interface agents, despite the admittedly spotty coverage and unreliable inference of today's commonsense knowledge systems. This article surveys several of these applications and reflects on interface design principles that enable successful use of commonsense knowledge.

Things fall down, not up. Weddings (sometimes) have a bride and a groom. If someone yells at you, they're probably angry.

One of the reasons that computers seem dumber than humans is that they don't have common sense—a myriad of simple facts about everyday life and the ability to make use of that knowledge easily when appropriate. A long-standing dream of artificial intelligence has been to put that kind of knowledge into computers, but applications of commonsense knowledge have been slow in coming.

Researchers like Minsky (2000) and Lenat (1995), recognizing the importance of commonsense knowledge, have proposed that common sense constitutes the bottleneck for making intelligent machines, and they advocate working directly to amass large collections of such knowledge and heuristics for using it.

Considerable progress has been made over the last few years. There are now large knowledge bases of commonsense knowledge and better ways of using it than we have had before. We may have become too used to putting common sense in that category of "impossible" problems and overlooked opportunities to actually put this kind of knowledge to work. We need to explore new interface designs that don't require complete solutions to the commonsense problem but can make good use of partial knowledge and human-computer collaboration.

As the complexity of computer applications grows, it may be that the only way to make applications more helpful and avoid stupid mistakes and annoying interruptions is to make use of commonsense knowledge. Cellular telephones should know enough to switch to vibrate mode if you're at the symphony. Calendars should warn you if you try to schedule a meeting at 2 AM or plan to take a vegetarian to a steak house. Cameras should realize that if you took a group of pictures within a span of two hours, at around the same location, they are probably of the same event.

Initial experimentation with using common sense encountered significant obstacles. First, despite the vast amount of effort put into commonsense knowledge bases, coverage is still sparse relative to the amount of knowledge humans typically bring to bear. Second, inference with such knowledge is still unreliable, due to vagueness, exceptional cases, logical paradoxes, and other problems.

Question-Answering Versus Interface Agent Applications

Many early attempts at applying common sense fell into the category of question-answering, story understanding, or information retrieval kind of problems. The hope was that use of commonsense inference would improve results beyond what was possible with simple keyword matching or statistical methods. See the sidebar "Other Applications of Commonsense Reasoning" (page 74) for some examples of these kinds of applications.

When direct question-answering using commonsense inference works, this is great. But direct question-answering places very exacting demands on a system.

First, the user is expecting a direct answer. If the answer is good, the user will be happy, if the answer is not, the user will be critical of the system. If accuracy falls below a certain threshold in the long term, the user will give up using the system completely. Second, the system gets only one shot at finding the correct answer, and it must do so quickly enough to maintain the feeling of interactivity (no more than a few seconds).

Over the last few years, we have been working in the area of intelligent interface agents (Maes 1994). An interface agent is an AI program that attaches itself to a conventional interactive application (text or graphical editor, Web browser, spreadsheet, and so forth), watches the user's interactions, and is capable of operating the interface as would the user. The jobs of the agent are to provide help, assistance, suggestions, automation of common tasks, adaptation, and personalization of the interface.

Our experience has been that interface agents can use commonsense knowledge much more effectively than direct question-answering applications can because they place fewer demands on the system. Because all the capabilities of the interactive application remain available for the user to use in a conventional manner, it is no big deal if commonsense knowledge does not cover a particular situation. If a commonsense inference turns out wrong, users are often no worse off than they would be without any assistance.

The user is not expecting a direct answer to every action, only that the agent will come up with something helpful every once in a while. Since the agent operates in a continuous, long-term manner, if it cannot respond immediately, it can gather further evidence and perhaps deliver a meaningful interaction in the future. If the agent's knowledge is not sufficient, it can ask the user to fill in the gaps.

In short, the use of common sense in interface agents can be made "fail-soft." Interface agents are often proactive, "pushing" information rather than "pulling" it as query-response systems do, and it is easier to make the former kind of agents fail-soft.

Applications of Common Sense in Interface Agents

The remainder of this article will survey several of our lab's recent projects in this area to illustrate the principles above. Except where noted, these applications were built using knowledge drawn from Open Mind Common Sense (OMCS) (see page 72), a commonsense knowledge base of over 725,000 natural language assertions built from the contributions of more than 15,000 people over the World Wide Web (Singh 2002). Many of these applications made use of early versions of ConceptNet, a semantic network of 1.6 million relations extracted from the OMCS corpus with 20 link types covering taxonomic, meronomic, temporal, spatial, causal, functional, and other kinds of relations.

We should note that all of the applications described below have the status of early prototypes. While none has had large-scale commercial deployment, many have fared well in small-scale user testing versus conventional applications. Details of testing are supplied in the references.

Common Sense in an Agent for Digital Photography

With the annotation and retrieval integration agent (Aria) (figure 1) (Lieberman, Rosenzweig, and Singh 2001), we attempt to leverage commonsense knowledge to semiautomatically annotate photos and proactively suggest relevant photos (Lieberman and Liu 2002). Aria observes a user as she or he types a story, parses the text in real time, and continuously displays a relevance-ordered list of photos. When the user inserts photos in text, the system automatically annotates the photos with relevant keywords. The basic Aria interface tested well in studies at the Kodak usability lab when compared with software shipped with Kodak cameras.

Commonsense knowledge is used in Aria to inform semantic recognition agents, which recognize people, places, and events in the text. These recognition agents extract appropriate annotations to be added to photos inserted in the text. In retrieval, commonsense knowledge is compiled into a semantic network, and

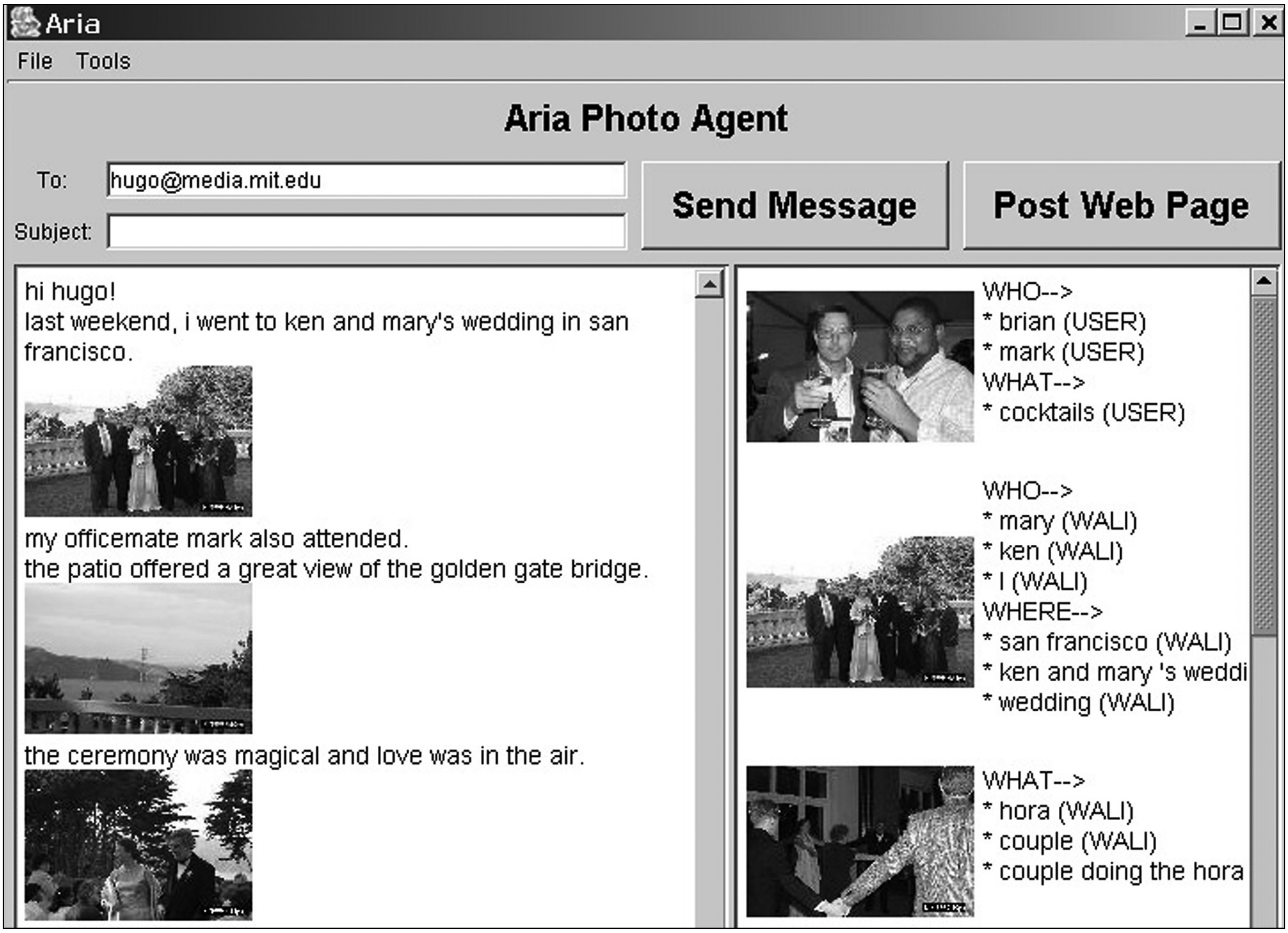


Figure 1. Telling Stories with ARIA.

associative reasoning helps to bridge semantic gaps (such as connect text about "wedding" to a photo annotated with "bride") (Liu and Lieberman 2002). The system also learns from personal assertions from the text (such as "My sister's name is Mary."), presumably unique to the author's context, which can be treated as a source of implicit knowledge in much the same manner as the commonsense assertions coming from Open Mind.

The application of common sense in Aria has several fail-soft aspects. Annotations suggested by the agent carry less weight than a user's annotations in retrieval and can be rejected or revised by the user. Similarly, in retrieval, common sense is used only to bridge semantic gaps and would never supersede explicit keyword matching. If a user finds a suggestion useful, she or he can choose to drag that photo in the text. But if the suggestion is inappropriate, the user's writing task is not disrupted.

Common Sense in Affective Classification of Text

Consider the text, "My wife left me; she took the kids and the dog." There are no obvious mood keywords such as "cry" or "depressed," or any other obvious cues, but the implications of the event described here are decidedly sad. This presents an opportunity for commonsense knowledge, a subset of which concerns the affective qualities of things, actions, events, and situations. From the Open Mind Common Sense knowledge base, a small society of linguistic models of affect was mined out, using a set of mood keywords as a starting point. The import of commonsense knowledge to this application is to make affective classification of text more comprehensive and reliable by considering underlying semantics, in addition to surface features.

Using this commonsense-informed ap-

Figure 2. Empathy Buddy Reacts to an E-mail.

proach, two applications were built. One is an e-mail editor, Empathy Buddy (figure 2), which uses Chernoff-style faces to interactively react to a user as she or he composes an e-mail using one of six basic Ekman emotions (Liu, Lieberman, and Selker 2003). A user study showed that users rated the affective software agent as being more interactive and intelligent than a randomized-face control.

Another application uses a hyperlinked color bar to help users visualize and navigate the affective structure of a text document (Liu, Lieberman, and Selker 2002). Using the tool, users were able to improve the speed of within-document information access tasks.

The affective model approach has been recently extended to modeling point-of-view and personality, analyzing an author's writings, and making a comparison of what several authors "might have thought" about a specified topic (Liu and Maes 2004).

Common Sense in Video Capture and Editing

The Cinematic Commonsense project (Barry and Davenport 2003) is being developed to provide feedback to documentary videographers during production. Commonsense knowledge relevant to the documentary subject domain is retrieved to assist videographers when they are in the field recording video footage. After each shot is recorded, metadata is created by the videographer in natural language and submitted as a query to a subset of the Open Mind database. For example, the shot metadata "a street artist is painting a painting" would yield a shot suggestion such as "the last thing you do when you paint a painting is clean the brushes" or "something that might happen when you paint a picture is paint gets on your hands." These assertions can be used by the filmmaker as a flexible shot list that is dynamically updated in accordance with the events the filmmaker is experiencing. Annotation of content is enriched, as in Aria, to support later search of image-based content. Collections of shots can also be ordered into rough temporal and causal sequences based on the associated commonsense annotations.

Initial tests with the system showed the need for more complex story understanding to create effective suggestions for the filmmaker. Currently, knowledge drawn from the Open Mind Experiences (Singh and Barry 2003) and LifeNet (Singh and Williams 2003) script acquisition systems (described later in this article) are being incorporated to enhance the reasoning ability of the system to accord with a few key documentary videographer goals, such as creating coherent narratives. In discussions with professional filmmakers, the prospect of tracking many potential narrative edits within a video database was seen as useful not only for closing gaps in content collections but also for illuminating alternative story ideas. This can encourage creative documentary in journalism, education, and everyday life.

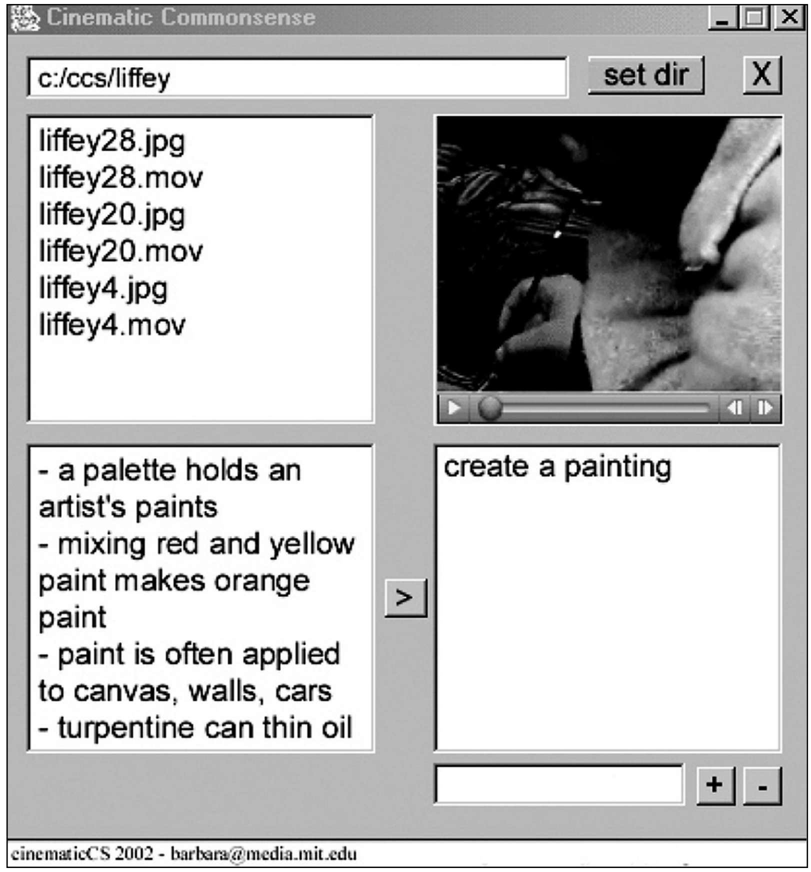


Figure 3. Common Sense Helps Associate Story Elements with Video Clips.

Common Sense in Other Storytelling Applications

A common thread throughout the aforementioned applications is that they all assist the user in some sort of storytelling process. Storytelling is a great area for common sense because it draws on a wide spectrum of understanding of situations of everyday life. It can provide an intermediate level for the agent to understand and assist the user that is better than simple keywords but stops short of full natural language understanding.

Alexandro Artola's StoryIllustrator[2] (figure 3) is like Aria in that it gives the user a story editor and photo database and tries to continuously retrieve photos relevant to the user's typing. However, instead of using an annotated personal photo collection, it employs Yahoo's image search to retrieve images from the Web. Commonsense knowledge is used for query expansion, so that a picture of a baby is associated with the mention of milk.

David Gottlieb and Josh Juster's OMAdventure[1] (figure 4) dynamically generates a Dungeons-and-Dragons type virtual environment by using commonsense knowledge. If the current game location is a kitchen, the system poses the questions to Open Mind, "What do you find in a kitchen?" and "What locations are associated with a kitchen?" If "You find an oven in a kitchen," we ask "What can you do with an oven?" Objects such as the oven or operations such as cooking are then made available as moves in the game for the player to make, and the associated locations are the exits from the current situation. If the player is given the opportunity to create new objects and locations in the game, that can be a way of extending the knowledge. If the player adds a blender to a kitchen, now we know that blenders are something that can be found in a kitchen.

Chian Chuu and Hana Kim's StoryFighter[3] plays a game where the system and the user take turns contributing lines to a story. The game proposes a start state, for example, "John is sleepy," and an end state, "John is in prison," and the goal is to get from the start state to the end state in a specified number of sentences. Along the way there are "taboo" words that can't be mentioned ("You can't use the word 'arrest'") as an additional constraint to make the game more challenging. Common sense is used to deduce the consequences of an event ("If you commit a crime, you might go to jail") and to propose taboo words to exclude the most obvious continuations of the story.

Common Sense for Topic Spotting in Conversation

Nathan Eagle, Push Singh, and Sandy Pentland

Welcome To OM Adventure

The Best Choose Your Own Adventure Game Available

Click an object in the room or
click magic to add something new
You are in the town
You are exploring corner grocery in the town
Are you adding a new object or a new location?
A fruit is always, often, sometimes, rarely or never in corner grocery
A fruit is always found in a corner grocery was added to OMCS
You are exploring town from the corner grocery
You are exploring traffic artery in the town
You are exploring town from the traffic artery

Figure 4. OMAdventure Dynamically Generates an Adventure Game's Universe by Using Commonsense Knowledge.

(Eagle, Singh, and Pentland 2003) are exploring the idea of a wearable computer with continuous audio (and perhaps ultimately, video) recording. They are interested not only in audio transcription, but in situational understanding—understanding general properties of the physical and social environment in which the computer finds itself, even if the user is not directly interacting with the machine.

Speech recognition is used to roughly transcribe the audio, but with current technology, speech transcription accuracy, especially for conversation, is poor. However, understanding general aspects of the situation such as whether the user is at home or at work, alone or with people, with friends or strangers, is indeed possible. Such recognition is vastly improved by using commonsense knowledge to map from topic-spotting words output by the speech recognizer ("lunch," "fries," "Styrofoam") to knowledge about everyday activities that the user might be engaged in (eating in a fast-food restaurant). Bayesian inference is used to rank hypotheses generated by ConceptNet.

Austin Wang and Justine Cassell used common sense in a virtual collaborative storytelling partner for children (Wang and Cassell 2003) whose goal is to improve literacy and storytelling skills. An on-screen character, SAM, starts telling a story and invites the child to continue the story at certain points. For example, "Jack and Jane were playing hide and seek. Jane hid in… now it's your turn."

The system uses speech recognition to listen to the child's story, but the recognition is not good enough to be sure of understanding everything the child had to say. Instead, the results of the recognition are used for rough topic-spotting, in the manner of Eagle, Singh, and Pentland's system.

In the hide and seek example, the system could hear the word "bedroom." Then commonsense knowledge is used to determine what is likely to be in a bedroom, such as a bed, closet, dresser, and so forth. The result is used to concoct a plausible continuation of the story when it is the virtual character's turn again to talk, for example, "Jane's parents walked into the bedroom while she was hiding under the bed."

Common Sense for a Dynamic Tourist Phrasebook

Globuddy (Musa et al. 2003), by Rami Musa, Andrea Kulas, Yoan Anguilette, and Madleina Scheidegger, uses common sense to aid tourists with translation. Phrasebooks like Berlitz will commonly provide a set of words and phrases useful in a common situation, such as a restaurant or hotel. But they can cover only a few such situations. With Globuddy, you can type in your (perhaps unusual) situation ("I've just been arrested") and it retrieves common sense surrounding that situation and feeds it to a translation service. "If you are arrested, you should call a lawyer." "Bail is a payment that allows an accused person to get out of jail until a trial." A recent implementation by Alex Faaborg and José Espinosa puts Globuddy on handheld and cellular telephone platforms (figure 5).

Common Sense for Word Completion

Applications like Globuddy play up the role of commonsense knowledge bases in determining what kinds of topics are "usual" or "ordinary." A simple, but powerful application of this is in predictive typing or word or phrase completion. Predictive typing can vastly speed up interfaces, especially in cases where the user has difficulty typing normally or is using small devices such as cellular telephones whose keyboards are small. Conventional approaches to predictive typing select a prediction either from a list of words the user recently typed or from an ordered list of the most commonly occurring words in English. Alex Faaborg and Tom Stocky (Stocky, Faaborg, and Lieberman 2004) have implemented a commonsense predictive text entry facility for a cellular telephone platform. It uses ConceptNet to find the next word that "makes sense" in the current context. For example, typing "train st" leads to the completion "train station" even though the user may not have typed that phrase before, nor is "station" the most common "st" word (figure 6).

Performance of common sense alone in this task is statistically comparable or slightly better than conventional frequency-based methods when measured with single-candidate prediction and may be much better on multiple-candidate prediction or when combined with conventional methods, especially where the conventional methods don't make strong predictions in particular cases. Similar approaches have great potential for use in other kinds of predictive and corrective interfaces.

Common Sense in a Disk Jockey's Assistant

Joan Morris-DiMicco, Carla Gomez, Arnan Sipitakiat, and Luke Ouko implemented a Common Sense Disk Jockey,[4] an assistant for music selection in dance clubs. Disk jockeys often select music initially based on a few superficial parameters (age, ethnicity, dress) of the audience and then adjust their subsequent choices based on the reaction of the audience.

CSDJ uses Erik Mueller's ThoughtTreasure as a reasoning engine (Mueller 1998) to filter a list of MP3 files according to commonsense assumptions about what kind of music particular groups might like. It also incorporates an interface to a camera that measures activity levels of the dance floor to give feedback to the system as to whether the selection of a particular piece of music increased or decreased activity.

Figure 5. The Globuddy 2 Dynamic Phrasebook Gives Translations of Phrases Conceptually Related to a Seed Word or Phrase.

Common Sense for Mapping User Goals to Concrete Actions

We also have worked on some projects incorporating commonsense knowledge into conventional search engines. These applications still maintain the "one-shot" query-response interaction that we criticized in the beginning as being less suited to commonsense applications than continuously operating interface agents. However, we apply the common sense in a fundamentally different way than conventional attempts to add inference to search engines. The role of common sense is to map from the user's search goals, which are sometimes not explicitly stated, to keywords appropriate for a conventional search engine. We believe that this process will make it more likely that the user would receive good results in the case where conventional keywords wouldn't work well, thereby making the interface more fail-soft.

Two systems, Reformulator (Singh 2002) and GOOSE (Liu, Lieberman, and Selker 2002) are commonsense adjuncts to Google.

Reformulator, like Cyc, does inference on the subject matter of the search itself. Our work in improving search engine interfaces (Liu, Lieberman, and Selker 2002; Singh 2002), is motivated by the observation that forming good search queries can often be a tricky proposition. We studied expert users composing queries (Liu, Lieberman, and Selker 2002) and concluded that they usually already know something about the structure and contents of pages they are expecting to find. After a little bit of search common sense is used to decide on the nature of the expected results, the chain of reasoning leading from the high-level search intent to query formation is usually very straightforward and commonsensical.

By contrast, novice users lack the experience in chain reasoning from a high-level search intent to query formation, so they often state their search goal directly. For example, a novice may often type "my cat is sick" into a search engine rather than looking for "veterinarians, Boston, MA" even though the chain of reasoning is very straightforward.

In this situation, there is an opportunity for a search engine interface agent to observe a novice user's queries. The agent attempts to infer the user's intent, and when it is detected

Figure 6. Common Sense Can Lead to Good Suggestions for Word Completion.

that a query may not return the best results, the agent can help to reformulate the query using search expertise and inferencing over commonsense knowledge, and opportunistically suggest "Did you mean to look for veterinarians in Boston, MA?" above the displayed results. In GOOSE, we were able to improve a significant number of queries made by novice users (figure 7). However, in that system, we still needed users to help the system by manually disambiguating the type of search goal. Our current work on automated disambiguation will allow us to develop an interface agent that does not interfere with the user's task at all and suggests a better query (appearing above the search results) only if it is able to offer a better one. This allows the interface agent to make use of common sense to improve the user experience in a fail-soft way. If common sense is too spotty to reformulate a query, no suggestion is offered.

Another application that also maps between users' goals and concrete actions is currently under development by Alex Faaborg, Sakda Chaiworawitkul, and Henry Lieberman for the composition of Web services.

In Tim Berners-Lee's proposed vision of the next-generation semantic web (Berners-Lee, Hendler, Lassila 2004), users can state high-level goals, and agent programs can scout out Web services that can satisfy those goals, possibly composing multiple services, each of which accomplishes a subgoal, without explicit direction from the user. For example, a request "Schedule a doctor's appointment for my mother within ten miles of her house" might involve looking up directories of doctors with a certain specialty; checking a reputation server; consulting a geographic server to check addresses, routes, or transit; synchronizing the mother's and doctor's schedules; and so on.

We fully concur with this vision. However, to date, most of the work on the semantic Web has focused on the formalisms, such as XML, OWL, SOAP, and UDDI, that will be used to represent metadata stored on the Web pages that will presumably be accessed by these agents. Little work is concerned with how an agent might actually put together semantic Web services to accomplish high-level goals for the user.

Looking at currently available and proposed Web service descriptions, we see that even if everyone agrees on the representation formalism, different services might ask for and return different kinds of information for the same services, and connecting them is still a task that now requires a human programmer to anticipate the form and structure of such services.

For example, a weather service might deliver a weather report given a Zip code. But if the user asked "What's the weather in Denver?" then something has to know how Zip codes are associated with cities. This is a job for common sense.

Common sense is used to compose Web services in a manner similar to the way it is used in GOOSE. User goals are obtained through two different interfaces; one that allows natural language statement of goals and another that provides a sidebar to a browser that proposes relevant services interactively as the user is browsing. ConceptNet is used to expand the user goal so that it can potentially match semantically related concepts that may appear in the Web

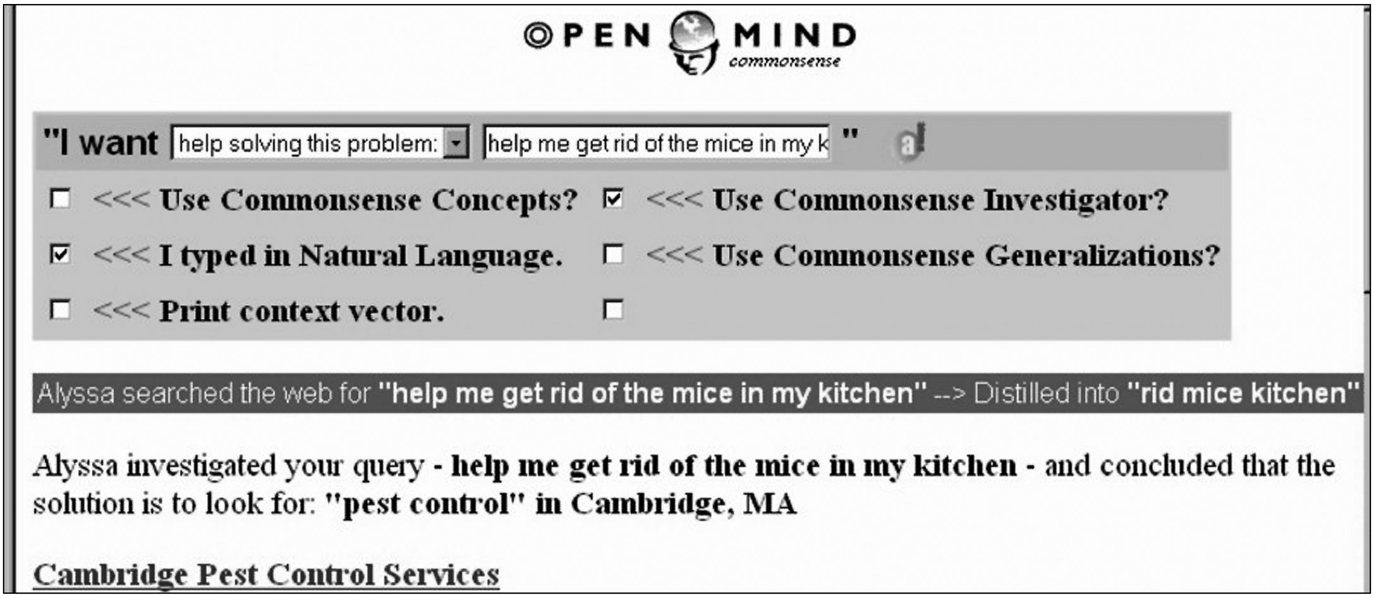


Figure 7. The GOOSE Commonsense Search Engine.

service descriptions. Thus we can achieve a much broader and more appropriate mapping of Web services than is possible with literal search through Web service descriptions alone.

Interfaces for Improving Commonsense Knowledge Bases

One criticism of Open Mind and similar efforts is that knowledge expressed in single sentences is often implicitly dependent on an unstated context. For example, the sentence "At a wedding, the bride and groom exchange rings" might assume the context of a Christian or Jewish wedding, and might not be true in other cultures. Rebecca Bloom and Avni Shah[5] implemented a system for contextualizing Open Mind knowledge by prompting the user to add explicit context elements to each assertion. Retrieval can then supply information about what context an assertion depends on or find analogous assertions in other contexts. For example, in a Hindu wedding, the bride and groom exchange necklaces that serve the same ritual function as rings do in the West.

Several projects involved interfaces for knowledge elicitation or feedback about the knowledge base itself. The Open Mind Web site itself contains several of what it calls "activities" that encourage users to fill in templates that call for a particular type of knowledge. Knowledge about the function of objects is elicited with a template "You __ with a __." Tim Chklovski developed an interface for prompting the user to disambiguate word senses in Open Mind (Chklovski and Mihalcea 2002) and for automatically performing simple analogies and asking the user to confirm or deny them (Chklovski 2003).

Andrea Lockerd's ThoughtStreams[6] aims to acquire commonsense knowledge through simulation. Everyday life is modeled in a game world, similar to the game, The Sims. An agent tracks user behavior in the world and tries to discover behavioral regularities with a similarity-based learning algorithm. It is also envisioned that a game character "bot" would be introduced that would occasionally ask human characters why they do things, in a manner of an inquisitive (but hopefully not too annoying) child.

Roles for Common Sense in Applications

Each of these applications uses common sense differently. None of them engage in "general-purpose" commonsense reasoning—while each makes use of a broad range of commonsense knowledge, each makes use of it in a particular way by performing only certain types of inferences. Also, while general-purpose logical inference is about what inferences are possible to make, commonsense inference is much more about what inferences are plausible to make. Our applications generally provide enough context about what is plausible so that we don't need to fall back on general logical mechanisms.

The Open Mind Common Sense Project

We built the Open Mind Common Sense (OMCS) Web site (openmind.media.mit.edu) to make it easy and fun for members of the general public to work together to build a commonsense database. OMCS was launched in September 2000, and as of November 2004 we have accumulated a corpus of about 690,000 pieces of commonsense knowledge from more than 14,000 people across the Web, many with no special training in computer science or artificial intelligence. Our top 100 list of contributors includes an artist, a chemist, a grandmother, a jockey, a 12-year-old child, a 911 police dispatcher, as well as people from many other walks of life. Our philosophy has been that because everyone has the common sense we wish to give our machines, everyone can contribute! The knowledge we have collected consists largely of the kinds of simple English assertions shown in table 1.

People live in houses.
Running is faster than walking.
A person wants to eat when hungry.
Things often found together: lightbulb, contact, glass.
Coffee helps wake you up.
Birds can fly.
The effect of going for a swim is getting wet.
The first thing you do when you wake up is open your eyes.

Table 1. Sample of OMCS Corpus

Our approach to knowledge acquisition was inspired by the success of information extraction methods (Cardie 1997). Rather than having our audience contribute knowledge directly in terms of some formal ontology, we instead encouraged them to express facts clearly in English via free-form and structured templates. Because they interacted with our system in plain English, they could begin entering information immediately—there was no extended training period where they had to learn a complicated knowledge-representation language. We extracted a semantic network from the resulting corpus of commonsense facts using a library of handcrafted lexico-syntactic pattern-matching rules (Liu and Singh 2004). This semantic network, called ConceptNet (www.conceptnet.org), consists of 280,000 links relating 300,000 concept nodes. The link types include a variety of common binary relations such as is-a, located-in, has-subevent, and so on. The concept nodes include a wide range of typical actions, objects, places, and properties, all expressed in terms of simple English phrases such as “go to restaurant” or “cup of coffee.” Because of its basis in natural language, the meanings of the links and nodes in ConceptNet are difficult to fully disambiguate. Nevertheless, ConceptNet has proven to be surprisingly useful and easy to use, especially in the context of “fail-soft” applications where the inferences do not always have to be correct.

Since the original Open Mind Common Sense Web site, there have been a number of projects that have taken new approaches to knowledge acquisition from the general public. The Open Mind Word Expert site (www.teach-computers.org) lets users tag the senses of the words in individual sentences drawn from both the OMCS corpus and the glosses of WordNet word senses (Chklovski and Mihalcea 2002). The Open Mind 1001 Questions site (www.teach-computers.org) uses analogical reasoning to pose questions to the user by analogy to what it already knows, and hence makes the user experience more interactive and engaging (Chklovski 2003). We developed the Open Mind Experiences site to allow users to teach stories in addition to facts, by presenting them with story templates based on Wendy Lenhert’s plot-units (Singh and Barry 2003). Finally, the upcoming LifeNet site lets users directly build probabilistic graphical models, and uses those models to immediately make inferences based on the knowledge that has been contributed so far (Singh and Williams 2003).

We are excited about these projects because it is now possible for AI researchers to work together and with the general public to build large-scale commonsense knowledge bases. Building practical commonsense reasoning systems is no longer solely the realm of multimillion dollar Manhattan projects and can be pursued in distributed and collaborative fashion by the AI community as a whole.

Retrieving Event-Subevent Structure

It is sometimes useful to collect together all the knowledge that is relevant to some particular class of activity or event. For example the Cinematic Common Sense project makes use of commonsense knowledge about event-subevent structure to make suitable shot suggestions at common events like birthdays and marathons. For the topic "getting ready for a marathon," the subevents gathered might include putting on your running shoes, picking up your number, and getting in your place at the starting line.

Goal Recognition and Planning

The Reformulator and GOOSE search engines exploit commonsense knowledge about typical human goals to infer the real goal of the user from their search query. These search engines can make use of knowledge about actions and their effects to engage in a simple form of planning. After inferring the user's true intention, they look for a way to achieve it.

Temporal Projection

The MakeBelieve storytelling system (Liu and Singh 2002) makes use of the knowledge of temporal and causal relationships between events in order to guess what is likely to happen next. Using this knowledge it can generate stories like David fell off his bike. David scraped his knee. David cried like a baby. David was laughed at. David decided to get revenge. David hurt people.

Particular Consequences of Broad Classes of Actions

Empathy Buddy senses the affect in passages of text by predicting only those consequences of actions and events that have some emotional significance. This can be done by chaining backwards from knowledge about desirable and undesirable states. For example, if being out of work is undesirable, and being fired causes to be to be out of work, then the passing "I was fired from work today" can be sensed as undesirable.

Specific Facts about Particular Things

Specific facts like "the Golden Gate Bridge is located in San Francisco," or "a PowerBook is a kind of laptop computer" are often useful. Aria can reason that an e-mail that mentions that "I saw the Golden Gate Bridge" meant that "I was in San Francisco at the time," and proactively retrieves photos taken in San Francisco for the user to insert into the e-mail.

Conceptual Relationships

A commonsense knowledge base can be used to supply "conceptually related" concepts. The Globuddy program retrieves knowledge about the events, actions, objects, and other concepts related to a given situation in order to make a custom phrasebook of concepts you might wish to have translations for in a given situation.

Limitations of the Role of Common Sense

We are well aware that there are limits to the use of common sense in applications. Again, these applications are early-stage prototypes and not large-scale fielded applications.

Among the principal problems we had in developing these systems was the spottiness of subject coverage of Open Mind. Open Mind knows a lot about certain things in some areas and little in others, as would be expected from a project that relies on volunteer labor. In some cases, we "beefed up" Open Mind's knowledge by deliberately entering facts we knew to be useful for a certain application. But we were happy to discover that the newly entered statements then "cross-fertilized" better behavior in some of the other applications, even when we weren't expecting it. As Open Mind–like projects scale up, these problems should decrease. All in all, we were more often surprised by the usefulness of what was in Open Mind than what it lacked.

We were limited by our inference methods. Inferences about events, actions, and objects as described above could not be counted on to be reliable and so were best used only when the situation was noncritical and inference chains were kept short. But we see the opportunity for better inference methods not simply by applying existing formal inference methods to commonsense knowledge but by developing new methods more suited to commonsense reasoning. We are exploring a method that interleaves context-sensitive inference steps with heuristic retrieval steps in a breadth-first manner. This will be the subject of a future paper.

We were also worried about the danger that commonsense-based suggestions might be distracting in the interface, since they compete for attention with more conventional interface elements. Happily, this did not turn out to be the case in user testing; We got strong positive reactions, and few users gave negative reactions that characterized the commonsense suggestions as a waste of time. Part of the reason was that, even when common sense doesn't lead to a correct conclusion, it often leads to a plausible conclusion. People are very much more tolerant of possibly correct wrong guesses

Other Applications of Commonsense Reasoning

Because broad-spectrum application of commonsense reasoning is still not widespread, we can't yet point to very many applications, especially in the commercial world, that use it. But we don't want to leave the impression that we at MIT are alone in pursuing this. Doug Lenat's CyCorp currently has the largest commonsense knowledge base and the longest experience in creating applications with it. Several Cyc applications were directed at trying to improve information retrieval beyond simple keyword matching by performing implicit inference.

For example, in a demonstration of Cyc (Lenat 1995), one could ask a news database "Show me a picture of someone who is disappointed" and receive a picture of the second finisher in the Boston Marathon, by a chain of reasoning like the following:

A marathon is a contest;

The goal of a contest is to be first;

If you do not achieve your goals, then you will be disappointed.

A subsequent project, eCyc, attached Cyc to the Lycos search engine, again to perform the same kind of implicit inference. This is somewhat like the way common sense is used in Liu's GOOSE project (Liu, Lieberman, and Selker 2002), except that in GOOSE the commonsense inference is used to infer the goal of the user's query (which may be not stated), whereas in eCyc, the commonsense inference is applied to the subject matter of the query, as it is expressed in the query.

On Cycorp's Web site (www.cyc.com), the security application CycSecure is described. It provides network vulnerability analysis and intrusion detection. It is not clear how, if at all, general knowledge about everyday life is used in this application as opposed to specific and precise technical information about networks. But common sense could be useful in tracking terrorist threats, for example, by noting that the word *Anthrax* probably refers to the disease in that context and not to the heavy metal band of that name.

Erik Mueller's SensiCal (Mueller 2000) is a "sanity checker" that provides warnings in a calendar scheduling program if it detects possible anomalies, such as scheduling a business meeting for 2 AM or planning to take a vegetarian friend to a steak house. It doesn't prohibit such actions, but asks for confirmation in the case that they were performed inadvertently. Cyc has also reported a sanity checker that is attached to a spreadsheet and is able to detect such problems as when a cell labeled as containing an age becomes less than zero.

Andrew Gordon (Gordon 2001) implemented a commonsense photo retrieval system that was tested in the Library of Congress. It uses a set of 768 "scripts" of everyday activities and relates these to the standardized vocabulary for annotating photographs. So, for example, a picture of *children* might have a related link to *students* (because the majority of students are children) in the annotation vocabulary, but Gordon's system can also supply *teachers,* because *students* and *teachers* are related in the *school* script. The interface was more of a typical information retrieval system than the real-time, proactive nature of our Aria.

There are also many projects that take a different and more formalistic slant to the commonsense problem, following the original idea of John McCarthy (McCarthy 1959). A recent overview can be found in Morgenstern and Davis (2004). Their approach is to isolate some particular capability of commonsense reasoning that they consider fundamental and attempt to do an exhaustive axiomatic analysis of it. Examples of domains that people have attacked in this manner are temporal reasoning or reasoning about the behavior of fluids. You could call this "formal reasoning about particular commonsense topics" as opposed to the "(informal) reasoning about common sense (as a whole)," which describes our approach here.

Obviously, when a formal approach is successful in identifying axioms and inference rules in a particular domain, it can replace hundreds or thousands of our Open Mind statements and reason with them more accurately. But each of these efforts is constrained to its domain, and it is not clear how many of these domains are necessary before what humans consider a reasonable scope of commonsense knowledge is attained. We prefer to work with a broad base of knowledge and see what can be done, even if the knowledge itself is shallow and the inferences inaccurate.

There are also several resources that provide broad-spectrum knowledge, but not what might be considered common sense in the sense of contingent knowledge about everyday life. For example, WordNet is a popular resource, used in many AI natural language programs, that provides word-sense disambiguation, hypernyms, and hyponyms. But its knowledge is mostly definitional. For example, Wordnet defines dog as a canine, which is a mammal. For ConceptNet, however, a dog is a pet, much more in line with the most salient feature for commonsense reasoning.

This also opens up broader consideration of what are termed knowledge-based applications in AI, including traditional expert systems, where the knowledge is expressed as rules. Many of these systems, also, operate in a commonsense domain, and thus could be said to be applications of commonsense knowledge. Like the applications produced by the formalists, they can have deep knowledge, but their scope is usually not very broad, and they are brittle outside their intended scope. The applications we have cited here, including the Cyc ones, Mueller's SensiCal, and Gordon's photographic system, share with our applications the property that they can talk about (almost) any subject that people would consider common knowledge, with a reasonable probability of doing something interesting.

than they are of guesses that are simply wildly off the mark.

Conclusions

We think that system implementers often fail to realize how underconstrained many user interface situations are. In many cases, systems either do nothing or perform actions that are essentially arbitrary. These applications show that there exists the potential to use commonsense knowledge to do something that at least might make sense as far as the user is concerned.

A little bit of knowledge is often better than nothing. Many applications, such as storytelling or language translation for tourists, can cover a broad range of subjects. With such applications, it is better to know a little bit about a lot of things than a lot about just a few things. Many past efforts have been stymied by insisting that coverage of the knowledge base be complete. They are often afraid to perform inferences because of the possibility of error. We rely on the interactive nature of the interface to provide feedback to the user and the opportunity for correction and completion.

Explicit input from the user is very expensive in the interface, so commonsense knowledge can act as an amplifier of that input, bringing in related facts and concepts that broaden the scope of the application.

Although our descriptions of each of these projects have been necessarily brief, we hope that the reader will be impressed by the breadth and variety of the applications of commonsense knowledge. We don't have to wait for complete coverage or completely reliable inference to put this knowledge to work, although as these improve, the applications will only get better. We think that the AI community ought to be paying more attention to this exciting area. After all, it's only common sense.

Notes

1. See web.media.mit.edu/~lieber/Teaching/Common-Sense-Course/Projects/OMAdventure/OMAdventure.doc

2. See www.media.mit.edu/~lieber/ Teaching/Common-Sense-Course/Projects/StoryIllustrator/StoryIllustrator-Intro.html

3. See web.media.mit.edu/~lieber/Teaching/Common-Sense-Course/Projects/Storyfighter/Storyfighter-Intro.html

4. See web.media.mit.edu/~joanie/commonsense/CSDJ-finalpaper.doc

5. See web.media.mit.edu/~lieber/Teaching/Common-Sense-Course/Projects/Contextualizer/Contextualizer.pdf

6. See www.media.mit.edu/~lieber/Teaching/Common-Sense-Course/Projects/ThoughtStreams/ThoughtStreams-Intro.html

References

Barry B.; and Davenport, G. 2003. Documenting Life: Videography and Common Sense. In *Proceedings of the 2003 IEEE International Conference on Multimedia and Expo*. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Berners-Lee, T.; Hendler, J.; and Lassila, O. 2001. The Semantic Web. *Scientific American* 284(5) (May): 28–37.

Cardie, C. (1997). Empirical Methods in Information Extraction. *AI Magazine* 18(4): 65–79.

Chklovski. T. 2003. LEARNER: A System for Acquiring Commonsense Knowledge by Analogy. Paper presented at the Second International Conference on Knowledge Capture (K-CAP 2003), Sanibel Island, FL, October 23–25.

Chklovski, T.; and Mihalcea, R. 2002. Building a Sense Tagged Corpus with Open Mind Word Expert. In *Proceedings of the Workshop on "Word Sense Disambiguation: Recent Successes and Future Directions."* East Stroudsburg, PA: Association for Computational Linguistics.

Davis, E.; and Morgenstern, L. 2004. Progress in Formal Commonsense Reasoning, Introduction to the Special Issue on Formalization of Common Sense. *Artificial Intelligence Journal* 153(1–2): 1–12.

Eagle, N.; Singh, P.; and Pentland, A. 2003. Common Sense Conversations: Understanding Casual Conversation using a Common Sense Database. Paper presented at the IJCAI-03 Artificial Intelligence, Information Access, and Mobile Computing Workshop, Acapulco, Mexico, August 2003.

Gordon, Andrew S. 2001. Browsing Image Collections with Representations of Commonsense Activities. *Journal of the American Society for Information Science and Technology* 52(11): 925–929.

Lenat, D. B. 1995. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM* 38(11): 33–38.

Lewin, D. B. 2000. Intelligencer: Instilling Common Sense into Software. *IEEE Intelligent Systems* 15(4): 2–6.

Lieberman, H.; and Liu, H. 2002. Adaptive Linking between Text and Photos Using Common Sense Reasoning. Paper presented at the Second International Conference on Adaptive Hypermedia and Adaptive Web Based Systems (AH2002), Malaga, Spain, May 29–31.

Lieberman, H.; Rosenzweig, E.; and Singh, P. 2001. Aria: An Agent for Annotating and Retrieving Images. *IEEE Computer* 34(7): 57–61.

Liu, H.; and Lieberman, H. 2002. Robust Photo Retrieval Using World Semantics. Paper presented at the Third International Conference on Language Resources and Evaluation Workshop: Using Semantics for Information Retrieval and Filtering (LREC2002), Las Palmas, Canary Islands.

Liu, H.; Lieberman, H.; and Selker, T. 2003. A Model of Textual Affect Sensing Using Real-World Knowledge. In *Proceedings of the 2003 International Conference on Intelligent User Interfaces*. New York: Association for Computing Machinery.

Liu, H.; Lieberman, H.; and Selker, T. 2002. GOOSE: A Goal-Oriented Search Engine with Commonsense. Paper presented at the Second International Conference on Adaptive Hypermedia and Adaptive Web Based Systems (AH2002), Malaga, Spain, May 29–31.

Liu, H.; Lieberman, H.; and Selker, T. 2002a. *Automatic Affective Feedback in an E-mail Browser,* Technical Report SA02-01, November 2002. Cambridge, MA: Massachusetts Institute of Technology Media Laboratory Software Agents Group.

Liu, H.; and Maes, P. 2004. What Would They Think? A Computational Model of Attitudes. In *Proceedings of the 2004 International Conference on Intelligent User Interfaces*. New York: Association for Computing Machinery.

Liu, H.; and Singh, P. 2004. Commonsense Reasoning in and over Natural Language. Paper presented at the Eighth International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES2004), Wellington, New Zealand, 22–24 September.

Liu, H.; and Singh, P. 2002. MAKEBELIEVE: Using Commonsense to Generate Stories. In *Proceedings of the Eighteenth National Conference on Artificial Intelligence* (AAAI-02), 957–958. Menlo Park, CA: AAAI Press.

Maes, P. 1994. Agents that Reduce Work and Information Overload. *Communications of the ACM* 37(7): 30–40.

McCarthy, John 1990. Computers with Common Sense. In *Formalization of Common Sense,* ed. V. Lipschitz. Westport, CT: Ablex Publishers.

Minsky, Marvin 2000. Commonsense-based Interfaces. *Communications of the ACM* 43(8): 67–73.

Mueller, Erik T. 2000. A Calendar with Common Sense. In *Proceedings of the 2000 International Conference on Intelligent User Interfaces,* 198–201. New York: Association for Computing Machinery.

Mueller, Erik T. 1998. *Natural Language Processing with Thought Treasure.* New York: Signiform.

Musa, R.; Kulas, A.; Anguilette, Y.; and Scheidegger, M. 2003. Globuddy, A Broad-Context Dynamic Phrasebook. In P*roceedings of the International Conference on Modeling and Using Context (CONTEXT '03), Lecture Notes in Computer Science*. Berlin: Springer-Verlag.

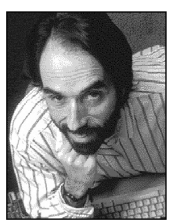
Singh, P. 2002. The Public Acquisition of Commonsense Knowledge. In *Acquiring (and Using) Linguistic (and World) Knowledge for Information Access: Papers from the 2002 AAAI Spring Symposium*, Technical Report SS-02-09. Menlo Park, CA: American Association for Artificial Intelligence.

Singh, P.; and Barry, B. 2003. Collecting Commonsense Experiences. Paper presented at the Second International Conference on Knowledge Capture (K-CAP 2003), Sanibel Island, FL, October 23–25.

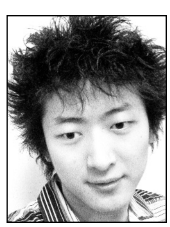
Singh, P.; and Williams, W. 2003. LifeNet: a Prepositional Model of Ordinary Human Activity. Paper presented at the Workshop on Distributed and Collaborative Knowledge Capture (DC-KCAP), Sanibel Island, FL, October 26, 2003.

Stocky, Tom; Faaborg, Alexander; and Lieberman, Henry 2004. Common Sense for Predictive Text Entry. Paper presented at the Conference on Human Factors in Computing Systems (CHI 2004), Vienna, Austria, April 24–29.

Wang, A.; and Cassell, J. 2003. Coauthoring, Collaborating, Criticizing: Collaborative Storytelling between Real and Virtual Children. Paper presented at the 2003 Vienna Workshop: Educational Agents—More than Virtual Tutors, Vienna, Austria, 20 July.



Henry Lieberman is a research scientist at the Massachusetts Institute of Technology's Media Laboratory. He heads the Software Agents Group, which applies AI techniques to making user interfaces more intelligent and easier to use. He edited the books Spinning the Semantic Web (MIT Press) and Your Wish Is My Command (Morgan Kaufmann). He is now working on common sense because he wishes he had more of it.



Hugo Liu is a Ph.D. candidate in media arts and sciences at the Massachusetts Institute of Technology's Media Laboratory. He has published more than a dozen expository papers on common sense and its application to the cognitive modeling of emotion, personality, hermeneutics, and information foraging. His research has been recognized with two best paper prizes. Hugo is presently engaged in the computational modeling of creativity, social identity, aesthetic experience, and the subconscious. He also essays irreverently on philosophy and postmodernism, and of course, he is Google-friendly. His e-mail address is hugo@media.mit.edu

Push Singh is a doctoral candidate in MIT's Department of Electrical Engineering and Computer Science. His research is focused on finding ways to give computers humanlike common sense, and he is presently collaborating with Marvin Minsky to develop an architecture for commonsense thinking that makes use of many types of mechanisms for reasoning, representation, and reflection. He started the Open Mind Common Sense project at MIT, an effort to build large-scale commonsense knowledge bases by turning to the general public, and has worked on incorporating commonsense reasoning into a variety of real-world applications. Singh received his B.S. and M.Eng. in electrical engineering and computer science from MIT.

Barbara Barry is a Ph.D. candidate at the Massachusetts Institute of Technology's Media Laboratory in the Interactive Cinema group. Her research interests include computational documentary, commonsense knowledge acquisition, and technologies for reflective practice. She earned a B.F.A. from Massachusetts College of Art and an M.S. from the MIT Media Lab. Contact her at barbara@media.mit.edu or visit www.media.mit.edu/~barbara.