Beating Common Sense into Interactive Applications

Henry Lieberman, Hugo Liu, Push Singh, Barbara Barry
MIT Media Lab, 20 Ames St., Cambridge, MA 02139 USA

Abstract

A long-standing dream of artificial intelligence has been to put *common sense* 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 common sense reasoning sufficiently robust.

Recently, at the MIT Media Lab, we have had some success in applying common sense knowledge in a number of intelligent Interface Agents, despite the admittedly spotty coverage and unreliable inference of today's common sense knowledge systems. This paper will survey several of these applications and reflect on interface design principles that enable successful use of common sense knowledge.

1 Introduction

Things fall down, not up. Weddings 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 common sense knowledge have been slow in coming.

Researchers like Minsky [2000] and Lenat [1995], recognizing the importance of common sense 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 common sense knowledge and better ways of using it then we have had before. We may have gotten 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 common sense 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 common sense knowledge.Cell phones 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 common sense 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.

2 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 common sense inference would improve results beyond what was possible with simple keyword matching or statistical methods.

For example, in a retrieval demo of Cyc [Lenat, 1995], one could ask "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: 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. When it 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 the accuracy falls below a certain threshold in the long term, the user will give up using the system completely. Second, the system only gets 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 exploring the domain 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, etc.) and both 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 common sense knowledge much more effectively than direct question-answering applications, because they place fewer demands on the system. Since all the capabilities of the interactive application remain available for the user to use in a conventional manner, it is no big deal if common sense knowledge does not cover a particular situation. If a common sense inference turns out wrong, the user is often no worse off then 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.

3 Applications of Common Sense in Interface Agents

The remainder of this paper 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 sidebar), a common sense knowledge base of over 675,000 natural language assertions built from the contributions of over 13,000 people over the World Wide Web [Singh *et al.*, 2002]. Many of these applications made use of early versions of *OMCSNet*, a semantic network of 280,000 relations extracted from the OMCS corpus with 20 link types covering taxonomic, meronomic, temporal, spatial, causal, functional, and other kinds of relations.

3.1 Common Sense in an Agent for Digital Photography

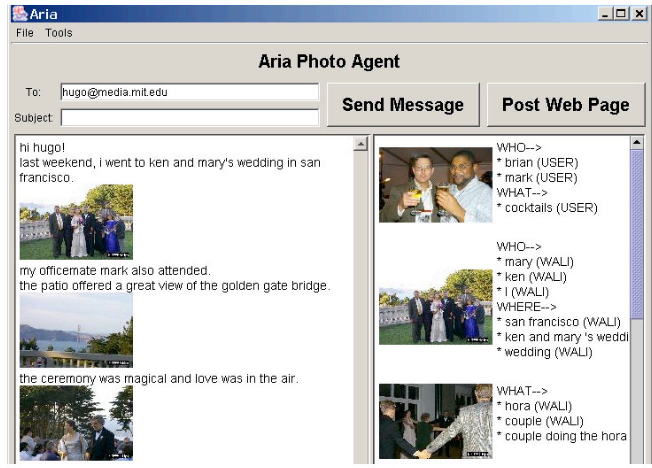

Figure 1. Telling stories with ARIA

In ARIA (Annotation and Retrieval Integration Agent, Figure 1) [Lieberman et al., 2001], we attempt to leverage common sense knowledge to semi-automatically annotate photos and proactively suggest relevant photos [Lieberman & Liu, 2002a]. ARIA observes a user as s/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.

Common sense knowledge is used 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, common sense knowledge is compiled into a semantic network, and associative reasoning helps to bridge semantic gaps (e.g. connect text about "wedding" to a photo annotated with "bride") [Liu & Lieberman, 2002b]. The system also learns from personal assertions from the text (e.g. "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 common sense 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, s/he can choose to drag that photo in the text. But if the suggestion is inappropriate, the user's writing task is not disrupted.

3.2 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 common sense 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 common sense knowledge to this application is to make affective classification of text more comprehensive and reliable by considering underlying semantics, in addition to surface features.

Figure 2. Empathy Buddy reacts to an email.

Using this commonsense-informed approach, two applications were built. One is an email editor, Empathy Buddy, above, which uses Chernoff-style faces to interactively react to a user as s/he composes an email using one of six basic Ekman emotions [Liu, Lieberman, 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, 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].

3.3 Common Sense in Video Capture and Editing

The Cinematic Common Sense project [Barry & Davenport, 2003] is being developed to provide feedback to documentary videographers during production. Common sense knowledge relevant to the documentary subject domain is retrieved to assist the videographer when they are in the field recording video footage about a documentary subject. 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 suggestions 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 be also ordered into rough temporal and causal sequences based on the associated common sense annotations.

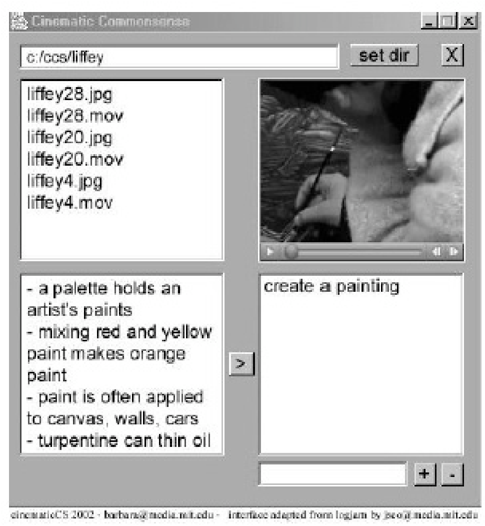

Figure 3. Common Sense helps associate story elements with video clips.

3.4 Common Sense in Other Storytelling Applications

A common thread throughout the above 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.

David Gottlieb and Josh Juster’s OMAdventure [Various Authors, 2003] (Figure 3) dynamically generates a Dungeons-and-Dragons type virtual environment by using common sense 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 are 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.

Figure 3. OMAdventure dynamically generates generates an adventure game’s universe by using common sense knowledge.

Alexandro Artola’s StoryIllustrator [Various Authors, 2003] (Figure 4) 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. Common sense knowledge is used for query expansion, so that a picture of a baby is associated with the mention of milk.

Chian Chuu and Hana Kim’s StoryFighter [Various Authors, 2003] plays a game where the system and the user take turns contributing lines to a story. The game proposes a start state, e.g. “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.

3.6 Common Sense for Topic Spotting in Conversation

Nathan Eagle, Push Singh and Sandy Pentland [Eagle, Singh, 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, etc., is indeed possible. Such recognition is vastly improved by using common sense 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 OMCS Net.

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's system.

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

3.7 Common Sense for a Dynamic Tourist Phrasebook

Globuddy [Musa et al., 2003], by Rami Musa, Andrea Kulas, Yoan Anguilete, 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 only cover 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 cell phone platforms.

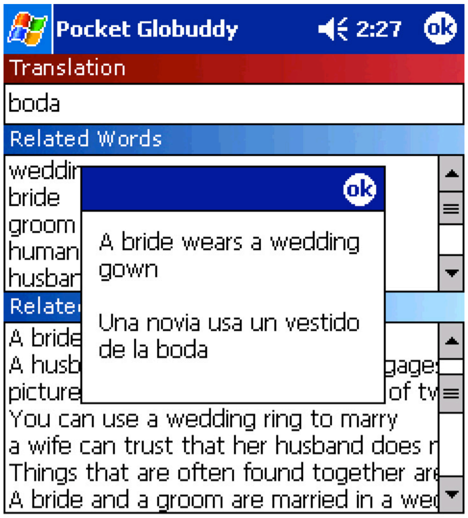

Figure 4. The Globuddy 2 dynamic phrasebook gives you translations of phrases conceptually related to a seed word or phrase

3.7 Common Sense for Word Completion

Applications like Globuddy play up the role of common sense 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 on small devices such as cell phones 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, Lieberman, 2004] have implemented a Common Sense predictive text entry facility for a cell phone platform. It uses Open Mind Common Sense Net 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.

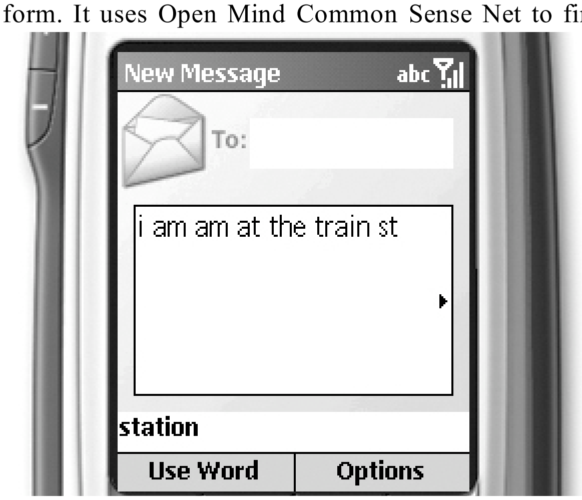

Figure 5. Common Sense can lead to good suggestions for word completion

Performance of Common Sense alone in this task is comparable or slightly better than conventional statistical methods and may be much better 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.

3.8 Common Sense in a Disk Jockey's Assistant

Joan Morris-DiMicco, Carla Gomez, Arnan Sipitakiat, and Luke Ouko implemented a *Common Sense Disk Jockey* [Various Authors], an assistant for music selection in dance clubs. DJs 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 common sense 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.

3.9 Common Sense for Mapping User Goals to Concrete Actions

We also have worked on some projects incorporating common sense 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 common sense 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 & Selker, 2002] are common sense 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 & 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, Liberman & 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 straight-forward 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 straight-forward.

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 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. 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 which does not interfere with the user's task at all, and only suggests a better query (appearing above the search results) 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.

Figure 6. The GOOSE common sense search engine

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; etc.

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. OMCSNet is used to expand the user goal so that it can potentially match semantically related concepts which may appear in the Web 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.

3.11 Interfaces for Improving Common Sense 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 [Various Authors, 2003] 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 [Chklovski & Mihalcea, 2002] developed an interface for prompting the user to disambiguate word senses in Open Mind and for automatically performing simple analogies and asking the user to confirm or deny them.

Andrea Lockerd's ThoughtStreams [Various Authors, 2003] aims to acquire common sense 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.

4 Roles for Common Sense in Applications

Each of these applications uses commonsense differently. None of them actually does '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.

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 common sense 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 common sense 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 & 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 prediction 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 "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 knowledgebase 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.

4.1 Do Try This at Home

We invite the AI community to make use of the Open Mind Common Sense knowledge base and associated tools to prototype applications as we have. We hope these application descriptions will inspire others to continue along these lines. Please see

http://openmind.media.mit.edu/.

We also welcome feedback from those who do choose to try this and would appreciate hearing of similar applications projects.

5 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 common sense 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 common sense 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 common sense 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.

Sidebar: Open Mind Common Sense

We built the the Open Mind Common Sense (OMCS) web site [http://openmind.media.mit.edu/] to make it easy and fun for members of the general public to work together to construct a commonsense database. OMCS was launched in September 2000, and as of January 2004 it has accumulated a corpus of about 675,000 pieces of commonsense knowledge from over 13,000 people across the web, many with no special training in computer science or artificial intelligence. The contributed knowledge is expressed in natural language, and consists largely of the kinds of simple assertions shown in Table 1.

Table 1. Sample of OMCS corpus

People live in houses.
Running is faster than walking.
A person wants to eat when hungry.
Things often found together: light bulb, contact, glass.
Coffee helps wake you up.
A bird flies.
The effect of going for a swim is getting wet.
The first thing you do when you wake up is open your eyes.
Rain falls from the sky.
Apples are not blue.
A voice is the sound of a person talking.

Rather than formulating a precise ontology in advance and then having knowledge enterers contribute knowledge expressed in terms of that ontology, we instead encouraged our users to provide information clearly in English via free-form and structured templates. Indeed, we sometimes think of OMCS not so much as a 'knowledge base' per se, but as a corpus of commonsense statements from which a more organized knowledge base can be constructed using information extraction techniques. In particular, we have extracted a large-scale semantic network called OMCSNet [Liu and Singh, 2004] consisting of 25 types of binary relations such as is-a, has-function, has-subevent, and located-in. The most recent version of OMCSNet contains 280,000 links relating 80,000 concepts, where the concepts are simple English phrases like 'go to restaurant' or 'shampoo bottle'.

We were surprised by the high quality of the contributions, given that the OMCS site had no special mechanisms for knowledge validation or correction. A manual evaluation of the corpus revealed that about 90% of the corpus sentences were rated 3 or higher (on a 5 point scale) along the dimensions of truth and objectivity, and about 85% of the corpus sentences were rated as things anyone with a high school education or more would be expected to know. Thus the data, while noisy, was not entirely overwhelmed by noise, as we had originally feared it might, and also it consisted largely of knowledge one might consider shared in our culture.

Several The Open Mind Word Expert site [http://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. The Open Mind 1001 Questions site [http://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. The Open Mind Experiences site [http://omex.media.mit.edu/] lets users teach stories in addition to facts by presenting them with story templates based on Wendy Lenhert's plot-units. Finally, the latest Open Mind 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.

References

[Barry & Davenport, 2003]. Barry B., and Davenport G. (2003). Documenting Life: Videography and Common Sense . In Proceedings of IEEE International Conference on Multimedia. New York: IEEE, 2003.

[Berners-Lee, Hendler, Lassila, 2004] Berners-Lee, T. Hendler, J., Lassila, O. The Semantic Web. Scientific American, May 2001.

[Chklovski & Mihalcea, 2002] Chklovski, T. and R. Mihalcea, (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", ACL 2002.

[Eagle, et. al, 2003] Eagle, N., P. Singh, A. Pentland, Common Sense Conversations: Understanding Casual Conversation using a Common Sense Database, Artificial Intelligence, Information Access, and Mobile Computing Workshop at the 18th International Joint Conference on Artificial Intelligence (IJCAI) Acapulco, Mexico. August 2003.

[Lenat, 1995] Lenat, D. B. (1995). CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11): 33-38.

[Lieberman & Liu, 2002a] Lieberman, H. and H. Liu, (2002). Adaptive Linking between Text and Photos Using Common Sense Reasoning. In Proceedings of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems, (AH2002) Malaga, Spain.

[Liu & Lieberman, 2002b] Liu, H. and H. Lieberman, (2002). Robust photo retrieval using world semantics. *Proceedings of the 3rd International Conference on Language Resources And Evaluation Workshop: Using Semantics for Information Retrieval and Filtering (LREC2002),* Las Palmas, Canary Islands.

[Liu, Liberman & Selker, 2002] Liu, H., Lieberman, H., Selker, T. (2002). GOOSE: A Goal-Oriented Search Engine With Commonsense. Proceedings of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems, (AH2002) Malaga, Spain.

[Liu and Maes, 2004] Liu, H. and P. Maes, (2004)., What Would They Think? A Computational Model of Attitudes. International Conference on Intelligent User Interfaces (IUI '04), January 2004, Funchal, Portugal.

[Lieberman *et al.,* 2001] Lieberman, H., E. Rosenzweig. P. Singh, (2001). Aria: An Agent For Annotating And Retrieving Images, *IEEE Computer*, July 2001, pp. 57-61.

[Liu *et al.,* 2003] Liu, H., H. Lieberman, , T. Selker, (2003). A Model of Textual Affect Sensing using Real-World Knowledge. In *Proceedings of IUI 2003.* Miami, Florida.

[Liu, Lieberman & Selker, 2002] Liu, H., H. Lieberman, and T. Selker, (2002) Automatic Affective Feedback in an Email Browser. MIT Media Lab Software Agents Group Technical Report SA02-01. November, 2002.

[Liu & Singh, 2002] Liu, H., P. Singh, (2002). MAKEBELIEVE: Using Commonsense to Generate Stories. In Proceedings of the 20th National Conference on Artificial Intelligence, (AAAI-02), 957-958, Edmonton, Canada

[Liu and Singh, 2004] Liu, H. and P. Singh (2004). The Open Mind Common Sense Net Toolkit. Draft at http://web.media.mit.edu/~hugo/publications/drafts/OMCSNet%20(CIKM).5.doc

[Maes, 1994] Maes, P. (1994). Agents that Reduce Work and Information Overload. *Communications of the ACM*, 37(7).

[Minsky, 2000] Minsky, Marvin (2000). Commonsense-based interfaces. *Communications of the ACM,* 43(8), 67-73.

[Mueller, 1998] Mueller, Erik T. (1998). Natural language processing with ThoughtTreasure. New York: Signiform. Available at: http://www.signiform.com/tt/book/

[Musa, et al., 2003] Musa, R., A. Kulas, Y. Anguilette, M. Scheidegger. (2003) Globuddy, A Broad-Context Dynamic Phrasebook, International Conference on Modeling and Using Context (CONTEXT '03), Stanford, CA., August 2003. Lecture Notes in Computer Science, Springer-Verlag, Heidelberg, 2003.

[Singh, 2002] Singh, P. (2002). The public acquisition of commonsense knowledge. In Proceedings of AAAI Spring Symposium: Acquiring (and Using) Linguistic (and World) Knowledge for Information Access. Palo Alto, CA, AAAI

[Stocky, Faaborg, Lieberman, 2004] Common Sense for Predictive Text Entry, submitted to CHI 2004. Vienna, April 2004.

[Various Authors, 2003] Various Authors (2003). Common Sense Reasoning for Interactive Applications Projects Page. http://www.media.mit.edu/~lieber/Teaching/Common-Sense-Course/Projects/Projects-Intro.html.

[Wang and Cassell, 2003] Wang, A. and J. Cassell, (2003). Co-authoring, Collaborating, Criticizing: Collaborative Storytelling between Real and Virtual Children, Vienna Workshop '03: Educational Agents - More than Virtual Tutors, Austrian Research Institute for Artificial Intelligence, Vienna, Austria, June 2003.