Collecting Commonsense Experiences

Push Singh
Media Lab, MIT
Cambridge, MA, 02139
push@media.mit.edu

Barbara Barry
Media Lab, MIT
Cambridge, MA, 02139
barbara@media.mit.edu

ABSTRACT

Humans naturally share knowledge by telling stories. This is a form of knowledge exchange we engage in right from early childhood, and over time we learn to recall, order and organize our experiences as stories [1]. In this paper we describe the Open Mind Experiences (OMEX) system, a web-based knowledge acquisition tool that exploits our natural ability to tell and explain stories in order to build a large-scale commonsense knowledge base. We built OMEX to gather descriptions and explanations of every-day, 'common sense' experiences from volunteer contributors distributed across the Internet. We first describe the results of our previous attempt to collect commonsense knowledge from the general public, the Open Mind Common Sense (OMCS) project. The OMCS project focused on collecting largely assertional commonsense knowledge, and we describe some of its products and spin-offs. We then give several motivating reasons for why we now wish to collect more script-like knowledge. We then explain the features of the new OMEX site and give an evaluation of system based on a preliminary user study. We conclude by discussing our future directions.

Categories and Subject Descriptors

I.2.4 Knowledge Representation Formalisms and Methods: *Frames and scripts*

I.2.6 Learning: *Knowledge acquisition*

General Terms

Design

INTRODUCTION

Humans naturally share knowledge by telling stories. This is a form of knowledge exchange we engage in right from early childhood, and over time we learn to recall, order and organize our experiences as stories [1]. Can we build a knowledge acquisition system that exploits our natural ability to tell and explain stories in order to build a large-scale commonsense knowledge base?

In this paper we describe the Open Mind Experiences (OMEX) system, a web-based knowledge acquisition tool we built to gather descriptions and explanations of every-day, 'common sense' experiences from volunteer contributors distributed across the Internet. We designed OMEX under the assumption that the contributors would have no background knowledge about artificial intelligence or computer science. Our goal is to accumulate a large database of descriptions of 'common sense' stories and explanations of these stories in structured sentences of the kind shown in Box 1, for there are presently no large-scale databases of structured story knowledge of this kind.

Stories:

Frank and Lily were on their way to a dinner party when they hit a pothole and got a flat. There was a spare in the trunk. They fixed the flat but arrived to the party very late.

Explanations:

Frank and Lily were in a car. A vehicle with wheels can get a flat tire. A flat tire prevents a vehicle from moving. People use cars to travel from one place to another. Parties are social gatherings of people. If a person is late they can miss part of an event. Changing a tire requires tools and takes about thirty minutes.

Box 1. An example of an explained story

This paper is organized as follows. We first describe the results of our previous attempt to collect commonsense knowledge from the general public, the Open Mind Common Sense (OMCS) project. The OMCS project focused on collecting largely assertional commonsense knowledge, and we describe some of its products and spin-offs. We then give several motivating reasons for why we now wish to now collect more script-like knowledge. We then explain the features of the new OMEX site and give an evaluation of system based on a preliminary user study. We conclude by discussing our future directions.

OPEN MIND COMMON SENSE

Can the general public really help us build large-scale commonsense knowledge bases? In this section we will briefly review our experience with the Open Mind Common Sense (OMCS) system [2,3], a web site designed to make it easy for members of the general public to contribute commonsense knowledge to a central database.

OMCS was built in the first half of the year 2000, inspired by the success of projects such as the Internet Movie Database[1] and the Open Directory Project[2], both enormous databases built by distributed communities of volunteers over the web. OMCS was launched in September 2000, and as of August 2003 it has accumulated a corpus of about 600,000 pieces of commonsense knowledge from over 11,000 people across the web, many with no special training in computer science or artificial intelligence. The contributed knowledge[3] 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.

To judge the quality of the contributions a manual evaluation of the corpus was performed [3]. This 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.

Our approach to knowledge acquisition in this project was based on the success of information extraction techniques in extracting information from raw text; see for example Cardie [4]. Rather than building an interface where users can directly engineer the knowledge representations used by the reasoning system, we instead encouraged them to provide information clearly in English via free-form and structured templates, and we later extracted semantic networks using simple information extraction methods. In particular, we extracted a large-scale semantic network called OMCSNet [5] consisting of 25 types of binary relations, e.g. **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'.

While the data we collected is noisy, it has nevertheless inspired and enabled us and others at our lab to experiment with building many new kinds of interactive applications, such as software agents for photo annotation and retrieval [6], web searching [7], topic spotting in spoken conversation [8], and inferring the affect of written text [9]. Enabling new kinds of applications was one of our goals at the outset of the OMCS project, for in our view we cannot study *practical* commonsense reasoning without some sort of commonsense knowledge base with knowledge of ordinary objects, events, locations, desires, jobs, relationships, and other kinds of concepts. Barry and Davenport [10] describe a system for generating and suggesting story threads, but that work has been hampered by the need for structured story knowledge about character goals, plan activation and termination criteria, and dependency of events in a story.

To use this data we have had to turn to alternative methods for commonsense reasoning; we had little success applying this data using traditional methods based on the application of logical rules of inference. Instead we have had some success employing 'weaker' methods of reasoning that nevertheless work over large quantities of noisy knowledge, for example, spreading activation along particular link types, or statistical reasoning where we add up evidence along different links. For example, our topic spotter can guess from a noisy transcription of a spoken conversation that the topic was 'eat in a restaurant' if its subevents contained words of the kind 'waiter', 'salad', 'food', and 'bill'. Our affective text classifier can, given a word like 'accident', follow links to consequences and post-events that had a clear affective association; the phrase 'getting into accident' would have the consequence of someone getting hurt, and hence the word 'accident' is assigned a negative valence. These methods are also fairly fast, and all of our applications so far work in real time on current hardware.

There is no doubt these kinds of techniques often produce incorrect inferences—for example, the emotional valence assigned to a given word in a sentence may be incorrect—but the kinds of applications that we have been exploring have largely been ones that are 'fail-soft' in the sense that the consequences of failure are relatively mild in the eyes of the user. These applications typically use commonsense reasoning to make suggestions or give the user feedback in a secondary window rather than serving as the bottleneck to getting a task done; question-answering, on the other hand, is an example of an application that is 'fail-hard' in that if the system produces a wrong answer the user is im-

[1] http://www.imdb.com

[2] http://dmoz.org

[3] This data is freely available for download at http://openmind.media.mit.edu/cgi-bin/download.cgi

mediately disappointed. Fail-soft applications give us a 'hill to climb' in that as our systems' abilities to reason improves, the systems should perform better, but even with weak reasoning they are still better than with no reasoning at all.

Since the development of the OMCS site, we have designed a next-generation site (OMCS-2) that corrected some of the problems of the original one, such as supporting knowledge validation and interactive analogical reasoning to give feedback to users [3]. This second site was partly built but never publicly launched. However, a few of its most important features appeared in later web sites. The Open Mind Word Expert site [11] lets users tag the senses of the words in individual sentences drawn from both the OMCS corpus and the glosses of WordNet [18] word senses. The Open Mind 1001 Questions site [12] 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 important lessons we learned from building the original OMCS site can be summarized as follows:

- There are many people out there willing to contribute to building a commonsense knowledge base. We obtained a substantial audience fairly quickly.
- The most useful knowledge was that supplied through templates, for these sentences were the easiest to map into semantic networks (by using the original template forms to code information extraction scripts.)
- The quality of the collected knowledge has been high enough to enable plausible inference based on spreading-activation or probabilistic reasoning, but still too noisy to support rule-based or logical reasoning.
- We do not need to wait until we have complete and perfect commonsense knowledge bases and inference techniques to begin exploring how to incorporate commonsense into applications.
- Simple reasoning techniques over large amounts of knowledge can aid 'fail-soft' applications where correct inferences are useful and incorrect inferences are not too much of a problem.

Hence we are confident that there is value to our approach, and we now wish to take it to the next, more interesting level. To us, the next step is collecting stories from people, as we will describe in the next section.

WHY COLLECT STORY KNOWLEDGE?

We have a number of motivations for collecting story knowledge.

We wish to enable a case-based approach to commonsense reasoning

Much work on giving machines common sense adopts a logical approach to commonsense reasoning. To a large extent the Cyc project [14] has proceeded according to this methodology, and it today consists of 1.5 million rules and facts in a logical language called CycL. Reasoning with Cyc happens largely through the application of various specialized theorem provers.

An alternative approach is to express commonsense knowledge in terms of larger frames or scripts that express concrete descriptions of particular objects, situations, and events [19, 20]. The important distinction between this approach and the logic-oriented approach is not so much one of expressivity—one can easily express frames and scripts in CycL, for example—but rather that one is expected to use frame and script knowledge via processes where the primary operations are about retrieving and adapting existing frames and scripts to new contexts, as opposed to the application of general rules of inference as is done in resolution theorem proving or tableaux methods. This approach is now known as case based or analogical reasoning, which employs a variety of mechanisms for retrieving, reusing, revising, and retaining cases [21, 22]. But despite the large amount of research to date on story understanding using frames and scripts, and on case-based and analogical reasoning in terms of concrete experiences, no system to date has been endowed with a large database of commonsense knowledge in the form of frames and scripts describing concrete experiences.

We wish to avoid having to formalize every commonsense domain in compact rules

While there is no doubt that it has been valuable and fruitful to model commonsense domains using logical rules, many commonsense domains are difficult to formalize in this way. For example, while there has been success modeling domains such as qualitative physics, where the underlying causal structure is relatively transparent, there has been less success formalizing more complex domains such as ordinary human social behavior. It is often argued that the strength of case-based reasoning is exactly in such hard-to-formalize domains. Case-based reasoning does not require a precise axiomatization of a domain for reasoning to proceed, because it possible to capture the flow of events without necessarily giving a deep theory of why those events played out the way they did, and case-based reasoning works by matching and adapting descriptions rather than reasoning from first principles. Avoiding brittleness, however, requires a substantial case library.

There are no substantial databases of script knowledge

None of the existing large-scale semantic knowledge bases contains a substantial amount of story knowledge. Mueller [13] compared several systems (Cyc [14], FrameNet [15], Gordon's Expectation Packages [16], ThoughtTreasure [17], and WordNet 1.6 [18]) and found that these systems consisted largely of facts and rules, and not cases and stories against which case-based reasoning could be performed. According to his review the most substantial corpus of script-like knowledge is Gordon's database of 768 'Expectation Packages', each which contains an average of 3.12 script steps. While the original OMCS system collected stories, most of what it collected was factual assertions similar to those in Cyc. While Cyc easily has the representational power to express concrete stories, it consists largely of 'general' knowledge in the form of small rules and facts rather than substantial descriptions of particular events.

Thus a large-scale story knowledge base would be a fundamentally new kind of resource.

Scripts implicitly contextualize knowledge

Some of the most challenging technical problems in building and using commonsense knowledge bases revolve around organizing and contextualizing the assertions [23, 24]. Cyc adopted the solution of putting each assertion within a well-defined context that captured the assumptions that underlie that assertion. In [24] Lenat argues that it is important to encode not just facts and rules about domains, but also meta-assertions that describe precisely in what situations those facts and rules apply, what other knowledge may be relevant, what problem-types those facts and rules may help solve, and so forth.

We regard stories as 'implicit contexts' for knowledge, which have many of the advantages of explicit contexts. A good story relates knowledge about, for example, the effects of an action (a flashlight lets you see in the dark) to problems such knowledge helps you solve (finding the car keys you dropped outside at night) to where and when such knowledge may be relevant (outside at night), and so forth.

It is easier for the general public to supply knowledge as stories

We suspect that the general public may be better at telling and explaining simple stories than directly formulating the rules that underlie a domain. While it is not hard to supply simple rules of the form "Eating poison will make you sick", it is hard to condition such assertions with the various caveats and circumstances which would allow the rules to apply more generally. It may be easier to tell and explain a specific story, which focuses the user on a specific set of characters, objects, and events, and their relationships, than to ask them to make a general rule-based theory in the abstract of some domain.

OPEN MIND EXPERIENCES

Because of these considerations, we decided that the next Open Mind web site should focus on gathering collecting descriptions everyday ordinary experiences. In this section we describe Open Mind Experiences (OMEX), and web site designed to collect such experiences from the general public. As of August 2003 this web site has not been publicly launched, but all of the features described here are implemented and are functioning in our lab.

After the user logs onto OMEX, they are presented with the search interface shown in Figure 1. This interface lets them search through and browse the stories that have been contributed so far. Rather than being organized as a large collection of separate activities, as in OMCS, this new site is organized as a search engine where the retrieved items can be individually operated on using the menu bar of operations listed above each contribution. Presently, the user can select among the following operations:

- *New*—Enter a new story of this general type.
- *Clone*—Start with a story exactly like this one, but

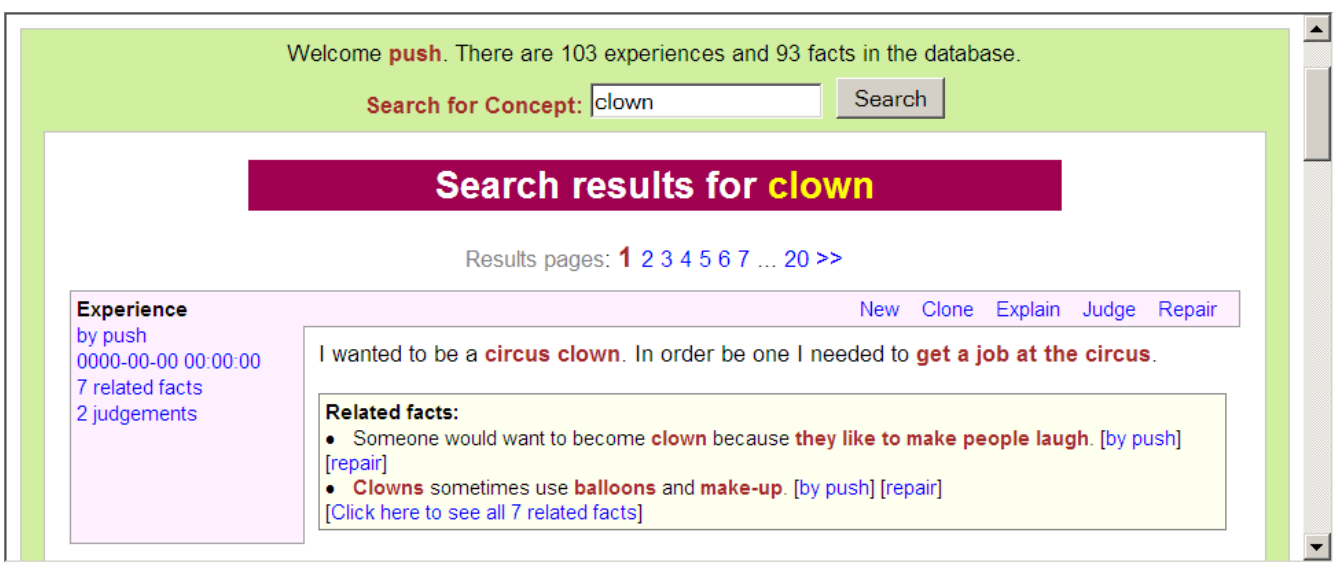

Figure 1. The main browser window

modify a few aspects.

- *Explain*—Explain this story by answering various questions about it.
- *Judge*—Evaluate this story along various dimensions.
- *Repair*—Suggest how to repair an error or other minor problem in a story.

We discuss each of these operations below.

New/Clone—Contribute a Story

The primary OMEX activity is selecting and filling in story templates. These templates were hand-built and are a thematic representation based on Lehnert's plot units [25]. Plot units are a convenient way to represent a wide range of story types. Plot units are graphs of linked positive events (+), negative events (-) and mental states (m). Lehnert used this representation as a way of identifying central concepts of a story plot during text summarization. We created templates based on plot units to prompt acquisition of stories across many subject domains.

Plot units are thematic story structures that are highly compositional. Each plot unit can be deconstructed into smaller units of events, actors and states; or, they can be aggregated into larger story structures. Plot units can represent simple themes involving one character, such as 'change of mind' or 'success' or more complex themes between multiple characters, such as 'retaliation' or 'competition'. The plot unit in Figure 2 shows competition between Frank and Lilly. A template would prompt the user to enter an identical mental event for Frank and Lilly, a positive event happening for Lilly and a negative event for Frank. The story contribution could be any of the following:

- Lilly and Frank both wanted the last bagel, but Lilly got it and Frank didn't.
- Frank and Lilly both wanted to be elected class president. Lilly won the election. Frank lost.
- Lilly and Frank were playing tennis. Lilly won the game.

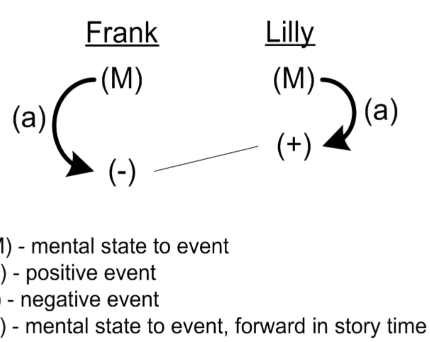

Figure 2. 'Competition' plot unit

Plot unit structures are the basis for template input design in OMEX. In the current version of OMEX there are 47 templates, each designed from a unique plot unit type. Our current set of templates is shown in Table 2.

Table 2. Current plot units

Motivation	Success
Failure	Change of Mind
Loss	Mixed Blessing
Perseverance	Resolution
Hidden Blessing	Enablement
Negative Trade Off	Complex Positive Event
Problem	Positive Trade Off
Complex Negative Event	Reneged-Promise
Initial-Problem-Resolution	Honored-Request
Ineffective-Coercion	Denied-Request
Fortuitous-Problem-Resolution	Bungled-Request
Success-Born-Of-Adversity	Effective-Coercion
Fleeting-Success	Bungled-Coercion
Ineffective-Coercion	Promised-Request-Honored
Promised-Request-Bungled	Double-Cross
Coerced-Agreement-Honored	Malicious-Act
Coerced-Agreement-Bungled	Kind-Act
Coerced-Double-Cross	Competition
Unsolicited-Help	Retaliation
Obligation	Regrettable-Mistake
Serial-Exchange	Sabotage
Simultaneous-Exchange	Problem-Resolution-by-Effective-Coercion
Honored-Promise	Request-Honored-With-Conditional-Request
Request-Honored-With-Conditional-Promise	

A screenshot of one such template is shown in Figure 3. The user fills in the blanks in the template with English phrases to complete the story.

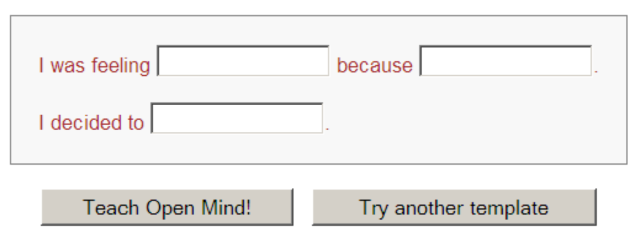

Figure 3. 'Motivation' template

In the future we will increase the number of templates per plot unit type, as we have found the template tends to influence the story content simply due to its syntactic form.

Explain—Explain a Story

Some knowledge is inconvenient to state explicitly in the story, for example, the causal relationships between the states and events. The Explain operation allows users to contribute the additional, supporting commonsense knowledge needed to understand the story. We distinguish between *general* and *specific* knowledge required to understand the story.

General Explanations

The Explain operation also allows the user to provide general commonsense knowledge in the form of simple assertions, as in OMCS. But unlike OMCS, the OMEX site collects such knowledge to provide the background knowledge needed to understand specific stories, as shown in Figure 4.

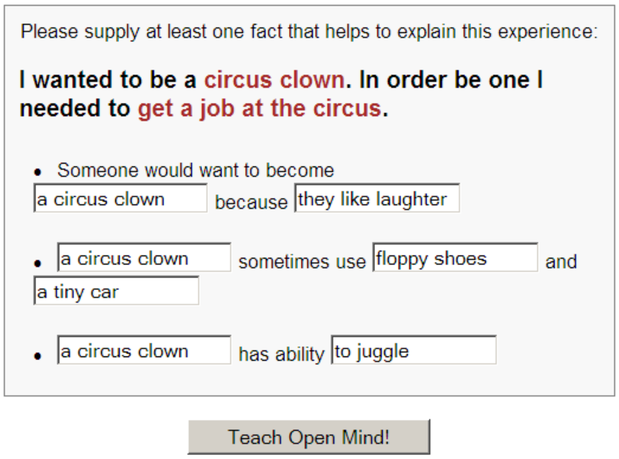

Figure 4. Explain a story

We see these contributed assertions as ‘implicitly contextualized’ by the story, for example, the sense of the word ‘pen’ in the contributed knowledge is likely to be the same as the sense used in the story.

OMEX presently has a set of 50 general ‘explanation types’ that it asks, for example:

- A property of *OBJECT that does not help its function is ?PROPERTY
- A broken *OBJECT can still be used to ?ACTION
- The *PROPERTY of *OBJECT enables it to ?ACTION
- A person needs to know how to *ACTION1 in order to ?ACTION2
- Something you need to do before *ACTION is ?ACTION2
- *OBJECT is usually found in a ?LOCATION
- *EVENT usually happens at a ?LOCATION
- *EVENT1 often immediately follows ?EVENT2

The variables beginning with an asterisk (*) are drawn from the feelings, events, and objects in the story being explained, and may be edited by the user to fix the inevitable syntax errors that result from substituting a phrase from one template into another template. The variables beginning with a question mark are left blank for the user to fill in.

Specific Explanations

The Explain operation also collects very particular assertions about a given story, for example:

- Does *PROPERTY of *OBJECT cause *EVENT?
- Did the event *EVENT1 cause the event *EVENT2?
- Can Jack see *OBJECT?

These are posed as yes/no questions to the user.

Judge—Evaluate this Story

The Judge operation lets the user assess individual stories along various dimensions, as shown in Figure 5.

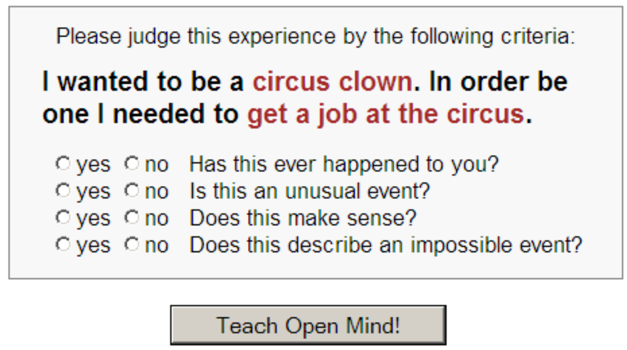

Figure 5. Judge a story

Our present set of judgments includes the following (not all shown in Figure 5):

- Is this typical or unusual event?
- Does this story describe an impossible event?
- Is this a story a 10 year old would understand?
- Is this an event that has ever happened to you?
- Does this story make sense?

We regard some of these judgments as criticisms that should lead to story being repaired, as described in the next operation.

Repair—Fix this Story

Knowledge bases built by a distributed community of volunteers require methods for assessing the quality of the knowledge that is contributed. The Repair operation allows users to suggest changes in items they consider problematic. There are two stages to a repair: Suggest Repair and Finalize Repair.

Suggest repair

When the user elects to repair an existing contribution a template of the erroneous item is presented as shown in Figure 6.

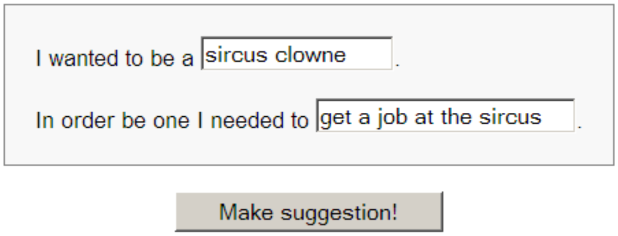

Figure 6. Suggest a repair

The user can make changes to any field of the template and then submit them for review. Changes can address spelling and grammatical errors as well as problems with the expression of the knowledge.

Finalize repair

A key difference from previous Open Mind sites is the introduction of an editorial board. The editorial board consists of a set of trusted users, and its role is to vet the suggested knowledge repairs that are made by the regular users of the site. Repaired items are presented to the editors for review. The editors can then make further changes to those items or immediately accept the modifications made by the users. Ultimately, the editors have final say over what changes are made to the database.

In future versions of the site we expect to add a mechanism for automatically suggesting promotions of regular users to the editorial board, based on the quality of their contributions so far.

EVALUATION

To give us a sense of the current OMEX system's performance, we performed a simple comparison between users using the web site and users simply writing stories of any length when prompted with a theme. The evaluation involved 10 people. 5 of those people worked for 15 minutes on the web site, and the other 5 worked for 15 minutes each writing stories and supporting fact knowledge in a text editor. A single judge ranked the resulting stories manually. Stories were ranked on a scale of 1-5 in the following categories: specificity (1=general description, 5=detailed instance), sense (1=makes no sense, 5=makes complete sense), typicality (1=would never happen, 5=happens all the time), age (1=young child, 5=senior), where we mean the age of someone who would understand the story. Figure 7 shows the average number of stories and supporting facts contributed by each group, those that used OMEX and those that used a text editor.

This shows that the OMEX site made it easy enough for people to rapidly enter stories, but our method for collecting single sentence assertions is slower. However, as with the OMCS site, we expect that the assertions collected through OMEX templates will be easier to parse and use than the free-form assertions supplied via the text editor. Also, as one might expect, when people used OMEX they tended to enter many variations on stories about the same subject, which suggests the contributions would lend themselves more easily to information extraction techniques.

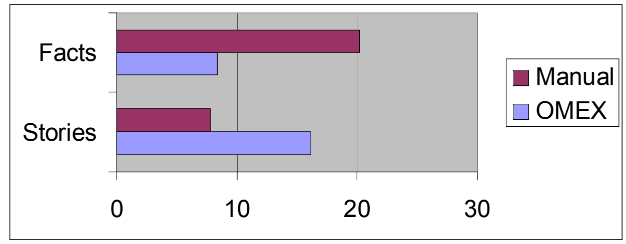

Figure 7. OMEX vs. Manual Text Editor

Figure 8 shows the evaluation of the data provided by the OMEX users.

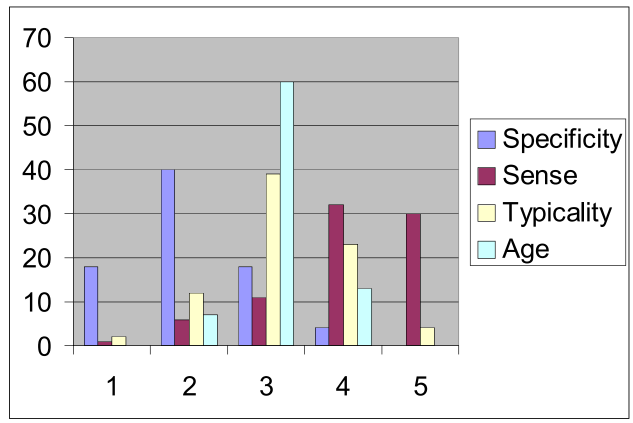

Figure 8. Evaluation of OMEX contributions

This shows about the stories we collect that: (a) they were fairly non-specific, in other words, lacking detail, probably because the templates encourage stories with simple event structures that do not allow supplying specific details about any given event; (b) they made sense, but when they did not it was due to the user fighting the syntactic restrictions of a template to tell they story they wished to tell; (c) they were moderately typical, which we interpret as meaning they were not anomalous events that would happen 'just once'—however, they may still not be typical enough given our goal of collecting common sense knowledge; and (d) they described experiences of people who were of college age—this is certainly because the authors of that experiences were themselves of college age.

In the future we plan to make the OMEX site somewhat 'self-evaluating' in the sense that judgments of the users about the collected stories will automatically generate graphs and charts of the kind depicted above, to provide the user community with a picture of the quality of the contributed knowledge.

FUTURE WORK

We are actively working on the following improvements to the system.

- Substantially increasing the number of story and assertion templates used by the system.

- Add a new operation called *Clarify* that will allow users to resolve word sense and anaphoric ambiguity within existing contributions.
- Generate new stories interactively with the system. Chklovski [12] found that posing new questions to users by analogy to existing knowledge increased the rate of knowledge entry on the Open Mind 1001 Questions site. A similar analogical mechanism could be employed to automatically generate plausible next steps for a story, which the user can accept or reject.
- Allow users to annotate stories with emerging standards for semantic annotation of text. For example, FrameNet and Cyc have both published standard sets of frame and slot types that could be used to mark up the collected stories.
- Produce a standard distribution of the collected knowledge analogous to our current OMCSNet distribution, in the form of an XML-based file format for the data, a client-side story browsing tool, and inference libraries supporting commonsense case-based reasoning.

Because so far we have tested the site primarily on people in our lab familiar with computers, and we have yet to test it on our target audience with a broader range of competency, it is likely that we will make further changes to the site before its launch, to simplify the interface, and possibly add help and tutorial pages.

CONCLUSIONS

In this paper we described the OMEX system for capturing story knowledge from the general public. The original OMCS web site attracted a substantial audience and we can only hope OMEX will do the same. Ultimately, our goal is to build a knowledge base with a million stories, which will be a unique new resource.

ACKNOWLEDGMENTS

We would like to thank William H. Williams for working with us on the earliest versions of OMEX, the members of the MIT commonsense reading group for beta testing recent versions and for many fruitful discussions about our general approach, Glorianna Davenport for her support and for sharing her expertise in interactive story construction, and Marvin Minsky for encouraging us to build a database of stories for commonsense reasoning. This project was supported by the sponsors of the MIT Media Lab.

REFERENCES

[1] Garvey, C. (1977). *Play*. Cambridge, Massachusetts: Harvard University Press.

[2] 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.

[3] Singh, P., Lin, T., Mueller, E. T., Lim, G., Perkins, T., & Zhu, W. L. (2002). Open Mind Common Sense: Knowledge acquisition from the general public. *Proceedings of the First International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems*. *Lecture Notes in Computer Science* (Volume 2519). Heidelberg: Springer-Verlag.

[4] Cardie, C. (1997). Empirical Methods in Information Extraction, *AI Magazine*, 65-79.

[5] Liu, H. and Singh, P. (2003). OMCSNet: A commonsense inference toolkit. Retrieved from http://web.media.mit.edu/~push/OMCSNet.pdf

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

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

[8] Eagle, N., Pentland, A., and Singh, P. (2003). Common Sense Conversations: Understanding Conversation using a Common Sense Data-base. To appear in *Proceedings of "Artificial Intelligence, Information Access, and Mobile Computing Workshop", IJCAI 2003*. Acapulco, Mexico. August 2003.

[9] Liu, H., Lieberman, H., and Selker, T. (2003). A Model of Textual Affect Sensing using Real-World Knowledge. In *Proceedings of the Seventh International Conference on Intelligent User Interfaces* (IUI 2003), pp. 125-132. Miami, Florida.

[10] Barry B. and Davenport G. (2003). Documenting Life: Videography and Common Sense. In *Proceedings of IEEE International Conference on Multimedia*. (ICME 2003), Baltimore, MD.

[11] 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", ACL 2002.* pp. 116-122.

[12] Chklovski, T. (2003). *Using Analogy to Acquire Commonsense Knowledge from Human Contributors*. MIT AI Lab technical report AITR-2003-002, February 2003.

[13] Mueller, E. T. (1999). *A database and lexicon of scripts for ThoughtTreasure*. CogPrints ID cog00000555 http://cogprints.soton.ac.uk/

[14] Lenat, D. B. (1995). CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, *38*(11), 33-48.

[15] Johnson, C. (1998). *Syntactic and semantic principles of FrameNet annotation*. University of California, Berkeley.

[16] Gordon, A. S. (1999). *The design of knowledge-rich browsing interfaces for retrieval in digital libraries*. Doctoral dissertation, Northwestern University, Evanston, IL.

[17] Mueller, E. T. (1998). *Natural language processing with ThoughtTreasure*. New York: Signiform.

[18] Fellbaum, C. (Ed.). (1998). *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.

[19] Minsky, M. (1974). *A framework for representing knowledge* (AI Laboratory Memo 306). Artificial Intelligence Laboratory, Massachusetts Institute of Technology.

[20] Schank, Roger C., and Abelson, Robert P. (1977). *Scripts, plans, goals, and understanding*. Hillsdale, NJ: Lawrence Erlbaum.

[21] Aamodt, A. and Plaza, E. (1994). Case-Based Reasoning: Foundational Issue, Methodological Variations, and System Approaches. *AI Communications*. IOS Press, Vol. 7: 1, pp. 39-59.

[22] Falkenhainer, B., Forbus, K.D. and Gentner, D. (1990). The structure-mapping engine: algorithm and examples. *Artificial Intelligence* 41, pp. 1-63.

[23] McCarthy, J. (1993). Notes on formalizing context. Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence.

[24] Lenat, D. (1998). The dimensions of context-space. Retrieved from http://www.cyc.com/context-space.doc.

[25] Lehnert, W. (1982). Plot Units: A Narrative Summarization Strategy. In Lehnert, W., and Ringle, M. (Eds.), *Strategies for natural language processing*. (pp. 375-414). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.