Open Mind Commons: An Inquisitive Approach to Learning Common Sense

Robert Speer
Commonsense Computing Group
MIT Media Lab
Building E15
77 Massachusetts Avenue
Cambridge, MA 02139-4307 USA
rspeer@mit.edu

ABSTRACT
Open Mind Commons is an interface for collecting common sense knowledge from users over the Web. By giving the user many forms of feedback and using inferences by analogy to find appropriate questions to ask, Open Mind Commons can learn well-connected structures of common sense knowledge, refine its existing knowledge, and build analogies that lead to even more powerful inferences.

ACM Classification: H.3.3 [INFORMATION STORAGE AND RETRIEVAL]: Information Search and Retrieval; I.2.6 [ARTIFICIAL INTELLIGENCE]: Learning

Keywords: Knowledge acquisition, commonsense, analogy, inference, feedback.

General terms: Languages, Human Factors

BACKGROUND
In many applications of artificial intelligence, interaction between the human and the computer is hindered by a fundamental gap in understanding between the user and the computer. The user is using the program to accomplish things in the real world; meanwhile, the computer doesn't even know what the real world is. The computer is lacking common sense, the body of basic knowledge that people know and computers don't.

In the Open Mind Common Sense project, we are collecting a corpus of common sense knowledge that we can use to make computers better understand real-world situations. Our approach is to harness the knowledge of large numbers of ordinary volunteers on the Web. The project's main Web site has collected over 700,000 statements from tens of thousands of volunteers [9].

Because the Web-based knowledge collection interface is the entry point for the vast majority of the knowledge in the system, improving the interface can help us collect higher-quality, more relevant knowledge. Open Mind Commons is the name for a new interface for knowledge collection, designed to help people work with the computer and with each other to create useful, well-connected knowledge structures.

Related Work
Some other projects that are confronting the task of common sense reasoning are Cyc [6], ThoughtTreasure [8], and Learner [3]. Cyc and ThoughtTreasure focus on reasoning by using a fine-tuned database of common sense that has been entered by knowledge engineering experts. Learner's approach is similar to Open Mind's; it, too, collects its knowledge from non-expert contributors over the Web.

The idea of making inferences by analogy and asking users to confirm them was originally implemented in Learner [3]. Open Mind Commons translates Learner's analogy approach to the ConceptNet representation used by Open Mind, and extends it so that analogies can build on other analogies.

ThoughtTreasure learns from analogy mostly in a different domain: it learns the meanings of unknown words by making analogies to the morphological derivations of words it knows [8]. This is an area that has not yet been explored in Open Mind, whose only use of morphological information is to assume that words with the same stem share the same meaning.

DESIGN
The Importance of Feedback
One complaint made by users of the original OMCS site was that there was no interesting feedback after the user entered a common sense statement. [11]. The item would simply vanish into the large database, giving no indication of how it helped the overall system learn or even whether it was helpful at all.

In Open Mind Commons, supplying interesting feedback to the user is a high priority in the design of the interface. It not only helps to retain the user's interest, it should lead to higher-quality, more relevant statements.

As the most prominent form of feedback, the system comes up with potential common sense inferences based on the in-

IUI'07, January 28–31, 2007, Hawaii, USA..

formation it already has on a topic, and asks the user to confirm them. This process serves many purposes: it confirms to the user that the system is understanding and learning from the data it acquires, it makes the database's knowledge in a given topic area more strongly connected, and it provides a way to check inferences for correctness, as common sense inferences can be unreliable. These inferences generally come from reasoning by analogy, and the process of making them is described below in the Inferences section.

The system also finds relevant fill-in-the-blank questions to ask the user using a very similar procedure: it simply finds inference candidates with one object left unknown. This yields relevant questions such as "You are likely to find ________ in a supermarket." This, too, helps to make the knowledge in the database more strongly connected.

Additional types of feedback that users can see include new inferences and analogies that have been made based on their contributions, ratings of their contributions by other users, and follow-up questions that the system asks after a user rejects a potential inference.

Dialogues with the User

When users interact with the inference engine, they can do more than just answer questions. The interface is intended to create a dialogue between the user and the computer, which can help the user understand the system better or help the system acquire richer knowledge.

After the computer asks a question based on an inference, such as "A bicycle would be found on the street. Is this common sense?", below it is a link labeled *Why do you ask this?* Clicking that link drops down the system's reason for believing that statement. In this case, the given justification is:

- A bicycle is similar to a car. (Why?)
- I've been told that a car would be found on the street.

Clicking the "Why?" link gives a further explanation of why the system concluded that a bicycle was similar to a car, by listing the statements it knows where "car" could be substituted for "bicycle":

- A bicycle is a vehicle.
- A bicycle is a mode of transportation.
- A bicycle is a machine.
- You are likely to find a bicycle in the garage.
- Bicycles can be used for transport.
- A bicycle has wheels.

Another kind of dialogue occurs when the user clicks "No" after the system asks a question based on an inference. This creates an opportunity for the system to learn much more than it would from an answer of "Yes", by asking the user to change an item to make the statement true.

As an example, the system may ask "Would you find a microwave in a bedroom?", based on a slight similarity it finds between a microwave and a bed (they're both household objects). The user clicks "No", which brings up the prompt for the user to alter the statement "You would find a microwave in a bedroom" to make it true.

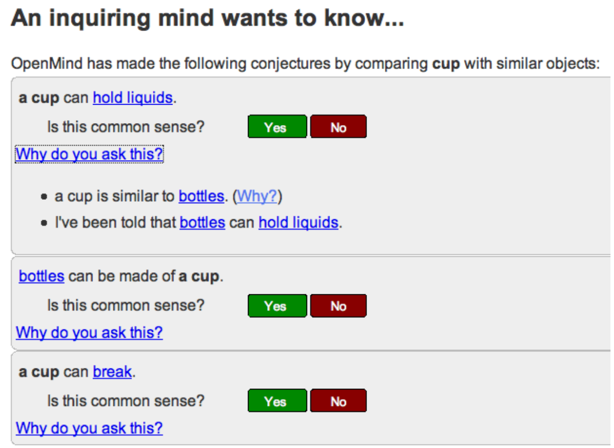

Figure 1: Open Mind Commons asks the user some questions to confirm its inferences.

If the user changes "a bedroom" to "a kitchen", then the system learns these new things:

- You would not find a microwave in a bedroom.
- You would find a microwave in a kitchen.
- A microwave isn't as similar to a bed as it previously seemed.
- A relevant difference between a microwave and a bed is that a microwave is found in a kitchen, while a bed is found in a bedroom.
- Because the user thought to type in "a kitchen", there may be a similarity between a kitchen and a bedroom.

The last two pieces of knowledge show the power of reasoning with *alignable differences*, a kind of data that has not existed in OMCS before Open Mind Commons.

Alignable Differences

An *alignable difference* is a pair of statements that describes different corresponding properties of two objects. One example of an alignable difference is "A hotel has many floors, while a motel has one or two floors". A non-alignable difference, in contrast, describes only a property that one object has and the other doesn't. An example of a non-alignable difference is "You can use a jacket to keep warm; you can't use a pencil to keep warm". People generally find that alignable differences are easier to think of and more meaningful than non-alignable differences [5]. The Open Mind Commons database only stores alignable differences, not non-alignable ones.

The system learns alignable differences mainly when the user corrects the results of its reasoning. In the situation described above, for example, "A microwave is found in a kitchen, while a bed is found in a bedroom" is an alignable difference that the system learns between a microwave and a bed. It could also be considered an alignable difference between a kitchen and a bedroom. Paradoxically, this difference tells the system to consider a microwave less similar to a bed than it did before, but also to consider a kitchen *more* similar to a bedroom. Why is this?

Every alignable difference needs to be built on similarity, because the objects involved need to be similar enough to have comparable properties. The fact that the original pair of objects (*bed* and *microwave*) were similar – in fact, the system considered them *too* similar – already played a role in learning this difference. But the other two objects (*bedroom* and *kitchen*) were paired *by the user*. The user is indirectly informing the computer that there is enough similarity between a bedroom and a kitchen to form an alignable difference, and the computer should take note by considering them more similar in the future.

This could be considered a form of the "near miss" learning proposed by Winston [12]. The system's guess of "You would find a microwave in a bedroom" is a near miss to the correct statement input by the user, "You would find a microwave in a kitchen". When comparing the target object *microwave* to the source object that created this analogy, *bed*, the important thing about the near miss is that it is a miss, but when comparing the newly introduced object *kitchen* to *bedroom*, the important thing about the near miss is that it is *near*.

IMPLEMENTATION

Knowledge Representation

Open Mind Commons uses a knowledge representation based on ConceptNet, which represents common sense knowledge as binary predicates that are closely tied to a representation in flat English text [7]. In this representation, English sentences are matched against a list of sentence patterns, each of which has two slots that can be filled by various phrases. These sentence patterns are mapped many-to-one onto relation types such as *IsA* and *EffectOf*. The phrases that fill the slots are converted to normalized forms by removing stopwords and running them through a stemmer. This process will convert, for example, the sentence "You are likely to find vegetables at a grocery store" to the internal representation *LocationOf*(*veget*, *grocery store*), which would also be the representation of "Something you find in a grocery store is a vegetable".

The representation used in Open Mind Commons builds on this by maintaining a reference to the words used in the original sentence. This lets inferred predicates be displayed to the user as coherent sentences, even though the stemmer discards suffixes and particles and produces non-words like *veget*. The stemmed version, which now comes from the Porter stemmer [10], is only used to compare statements, never to output them to the user.

Each predicate additionally has a rating, which is assigned by users of the site. Each user can increase or decrease a statement's rating by one point. The rating is mapped nonlinearly to a *weight* that determines how much the statement contributes to an inference: a statement with a negative rating has zero weight, because it can't be trusted, and a statement with a non-negative rating has a weight that grows logarithmically with the rating.

The new representation also includes a truth value for each predicate, so that negative statements such as "You would not find a microwave in a bedroom" can be represented. The truth value field currently is only used for distinguishing positive statements from negative, but it could be extended to a range of possible truth values to represent modifiers such as "sometimes", "rarely", or "almost always", once this is supported by the natural language processing and reasoning components. Although both are represented numerically, the truth value is quite distinct from the score: a reliable negated statement should have a positive score but a negative truth value.

Score	Relation	Item 1	Item 2
1640	CapableOf	fish	float in the water
1436	CapableOf	a person	die
1316	CapableOf	cat	eat
1268	CapableOf	a person	believe in god
1176	LocationOf	a car	on a table
1173	CapableOf	a person	read books
1114	LocationOf	a paper	in a room
1026	LocationOf	a car	the office
1012	LocationOf	students	a desk
987	CapableOf	a person	see with their eyes

Figure 2: The ten highest-scored inferences in Open Mind Commons, as of November 27, 2006. Items are shown in their unstemmed form.

These predicates are stored in a relational database, where SQL queries can be used to quickly make new inferences so that they can be presented to the user in real time. This database was initially populated with the English content of the GlobalMind project [4], a subset of ConceptNet that can easily be converted to this representation. The upcoming CSAMOA architecture [1] will make it possible to convert the original corpus of sentences contributed to OpenMind into the Open Mind Commons representation, at which point many more previously contributed common sense statements can be imported into the database.

Structure Mapping

Open Mind Commons makes the inferences that let it decide what questions to ask by making analogies to its existing knowledge, in a process inspired by the "structure mapping" process proposed by Gentner and Markman [5].

Two objects, or phrases that fill the slots of a template, are *similar* if they participate in many of the same relations. For example, the parallel statements *Cars can be used for transport* and *Bicycles can be used for transport* contribute to the similarity between *car* and *bicycle*.

The similarity score between objects A and B is calculated as

$$\mathrm{sim}(A,B) = \sigma(A,B) + k \cdot (\Delta^{+}(A,B) - \Delta^{-}(A,B))$$

Here $\sigma(A,B)$ is the weighted total number of pairs of statements where A can be substituted for B: that is, either $F(A,X)$ and $F(B,X)$ are both known, or $F(X,A)$ and $F(X,B)$ are both known. $\Delta^{-}(A,B)$ is the weighted number of alignable differences that count against the similarity between A and B, while $\Delta^{+}(A,B)$ is the weighted number of alignable differences that imply more similarity between A and B. (As described before, this depends on whether the similarity between A and B was used in learning the alignable difference, or whether it came from the user's input on the other side of the alignable difference.) k is a constant that determines the relative importance of similarities and differences.

To make analogies, the system considers a particular object A as the topic. It looks up which objects B_i are similar to A, then finds statements where B_i could be changed to A to yield a currently unknown statement. The analogies that are used are the ones that achieve the highest score, according to:

$$\text{inferscore}(F(A,C)) = \sum_{B_i} \text{sim}(A, B_i) \cdot \text{weight}(F(B_i, C))$$

Statements that are already known to be true or false, however, are never inferred.

Once a statement $F(A,C)$ is inferred by an analogy starting from A, the system confirms the inference by asking a user who browses to topic A if the statement is true. If the user answers *yes*, then the system learns the inferred statement as a new fact. But if the user answers *no*, the system takes this opportunity to learn more by asking a follow-up question. The system asks the user to change item C, if possible, to an item D that makes the statement true. If the user does so, then the system learns an alignable difference between $F(A,D)$ and $F(B_i,C)$ for each B_i. This causes an increase in both $\Delta^-(A,B_i)$ and $\Delta^+(C,D)$, and this increase feeds back into the similarity score, where it can be used to make further analogies. In this way, each alignable difference removes similarity from one pair of objects while adding it to another, in a process that can build on itself as the system learns.

FUTURE WORK

Multilingual Common Sense

The next step for the Open Mind Commons project is to integrate it with GlobalMind [4], a project for collecting common sense knowledge from various cultures in multiple languages. The ideas behind GlobalMind are that what is common sense for one group of people may not be for another, and that common sense is best expressed in one's native language.

GlobalMind encourages contributors to translate sentences between languages that they are familiar with, and uses the information provided by those translations to align the knowledge bases of different languages. As the knowledge representation of Open Mind Commons is compatible with that of GlobalMind, these features will be incorporated into Open Mind Commons in the near future.

GlobalMind currently contains knowledge in English, Korean, Japanese, and Chinese. A group in Brazil, meanwhile, has created a sister project to the original OpenMind that collects knowledge in Portuguese [2], and this knowledge base will also be included in Open Mind Commons.

In the same way that Open Mind Commons currently encourages well-connected knowledge structures within a knowledge base, it will be able to strengthen the connections between knowledge bases in different languages provided by GlobalMind.

Cultural comparisons

The ability in the existing GlobalMind project to align knowledge bases allows it to be used to find the places where knowledge can't be aligned – that is, where speakers of different languages have different views of the world. I believe that many more interesting and informative cultural differences in common sense can be found by comparing the contributions of different groups of people in the *same* language.

One study has already shown the potential of intercultural comparisons, by comparing statements from OpenMind contributors who identified themselves as being from the USA, Mexico, or Brazil. The statements were all either collected in English, or collected on the Portuguese-language site and translated to English. A structure-mapping process found relevant alignable differences between the cultures in mealtimes and in which foods are considered appropriate for various meals and occasions [2].

To enable cultural comparisons, Open Mind Commons will allow users to describe themselves, and then will use the contributed knowledge of different people to compare the common sense of different groups. These groups do not have to be limited to nationalities or speakers of a common language – comparisons can also be drawn between contributors of different age groups, sexes, occupations, or based on any other information that contributors are willing to provide.

A process similar to the inference process can then find similarities, alignable differences, and analogies between the views of different groups of people. The results can be applied in education and facilitating communication between people, and will form a more realistic model of human knowledge than one which assumes that everyone believes the same things.

REFERENCES

1. Alonso, J. 2007. CSAMOA: a common sense application model of architecture. IUI'07, January 28–31, 2007, Honolulu, Hawaii.

2. Anacleto, J. et al. 2006. Can common sense uncover cultural differences in computer applications? IFIP World Computer Conference, Santiago, Chile.

3. Chklovski, T. 2003. Learner: a system for acquiring commonsense knowledge by analogy. In *Proceedings of the 2nd international Conference on Knowledge Capture* (Sanibel Island, FL, USA, October 23 - 25, 2003). K-CAP '03. ACM Press, New York, NY, 4-12.

4. Chung, H. 2006. GlobalMind — bridging the gap between different cultures and languages with commonsense computing. MS Thesis.

5. Gentner, D. and Markman, A. B. 1997. Structure mapping in analogy and similarity. American Psychologist, 52, 45-56.

6. Lenat, D. B. 1995. CYC: a large-scale investment in knowledge infrastructure. Commun. ACM 38, 11 (Nov. 1995), 33-38.

7. Liu, H. and Singh, P. 2004. ConceptNet: a practical commonsense reasoning toolkit. *BT Technology Journal*, 22(4):211-226.

8. Mueller, E. T. 1998. Natural language processing with ThoughtTreasure. New York: Signifonn.

9. Open Mind Common Sense. http://openmind.media.mit.edu.

10. Porter, M. F. 1980. An algorithm for suffix stripping. Program, 14: 130-137.

11. Singh, P. et al. 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.* Irvine, CA.

12. Winston, P. H. 1975. Learning structural descriptions from examples. In Winston, P. H., editor, The Psychology of Computer Vision. McGraw-Hill.