

Learner: A System for Acquiring Commonsense Knowledge by Analogy

Timothy Chklovski

USC Information Sciences Institute
Marina del Rey, CA, USA
timc@isi.edu

ABSTRACT

One of the long-term goals of Artificial Intelligence is construction of a machine that is capable of reasoning about the everyday world the way humans are. In this paper, I first argue that construction of a large collection of statements about everyday world (a repository of commonsense knowledge) is a valuable step towards this long-term goal. Then, I point out that volunteer contributors over the Internet — a frequently overlooked source of knowledge — can be tapped to construct such a knowledge repository. To operationalize construction of a large commonsense knowledge repository by volunteer contributors, I then introduce *cumulative analogy*, a class of analogy-based reasoning algorithms that leverage existing knowledge to pose knowledge acquisition questions to the volunteer contributors. The algorithms have been implemented and deployed as the LEARNER system. To date, about 3,400 volunteer contributors have interacted with the system over the course of 11 months, increasing a starting collection of 47,147 statements by 362% to a total of 217,971. The deployed system and the growing collection of knowledge it acquired are publicly available from <http://teach-computers.org>. The knowledge is beginning to be used in follow-on research to address some well-recognized and novel reasoning tasks.

Categories and Subject Descriptors

I.2.6 [Learning]: Analogies, Concept learning, Knowledge acquisition

Keywords

Analogy, commonsense knowledge, volunteer contributors, natural language

1. INTRODUCTION

1.1 The case for creating a large commonsense knowledge repository

A currently unattained long-term goal of Artificial Intelligence is construction of computer programs capable of reasoning about everyday objects, events, and situations similarly to the way humans do. Authors of [13, 20, 5] argue that constructing software systems that approach the level of human reasoning about our world is impossible without constructing large knowledge repositories that contain statements about the world. For example, to act in the world or think about actions in the real world, one (a human or a program) would need such knowledge as: “rain falls from above,” “talking over loud noise is difficult,” “locked doors require keys,” and so on.

A typical human adult may have an enormous amount of commonsense knowledge at his disposal. The amount can be estimated to be at least several million axioms. Exact estimates vary; I briefly present several sources. Considering only the subset of the real-world facts relevant to medicine, an average person without medical training knows approximately 100,000 such facts, according to [23]. The amount of knowledge necessary for human-level commonsense reasoning has been estimated in [14] to be on the order of 10^8 axioms. Perhaps the most convincing bound comes from the studies of the amount of information humans are able to remember and retrieve after a long period. Landauer points out a remarkable constancy of human ability to memorize various types of information across a variety of modalities. Quantitative analysis across visual, verbal, and musical information yields similar results of humans being able to retain for the long term approximately two bits per second [12]. This implies a per-year rate of about 8 megabytes, or approximately a million short statements per year. This estimate does not include non-learned information such as pre-wiring for spatial or causal reasoning that may be encoded in the genome. Still, the total amount of statements humans are able to acquire seems to be at least several million axioms.

Much of current work on common sense reasoning ex-

plores issues of formal representation of knowledge (such as [18, 8] as well as work reported in the ongoing series of International Symposia on Logical Formalizations of Commonsense Reasoning). The focus of the work reported here is instead on acquiring commonsense knowledge: to address collection of the millions of statements that comprise common sense in support of further exploration of reasoning over such a knowledge repository and to enable application of commonsense knowledge to practical problems.

While some approaches have been to create such a collection of knowledge with a dedicated multi-year knowledge engineering effort (namely, CYC, [13]), or via mining text (e.g., [22]), I explore an alternative approach of getting such high volume of assertions. This approach is to acquire knowledge from many contributors by relying on analogical reasoning.

1.2 Collection of commonsense knowledge can be distributed across many contributors

A large repository of commonsense knowledge is an important resource for validation of the more formal explorations of commonsense reasoning; such a repository also holds the promise of enabling significant breakthroughs on many open problems in natural language processing and information retrieval, such as information extraction, question answering [22, 13], and machine translation.

To compare in size with the knowledge readily available to a human, estimates indicate that the KB has to contain at least several million axioms. Even ignoring the effort of verifying the quality of and of structuring such a repository, and assuming that only the needed axioms are entered, collecting this many axioms is a singularly labor-intensive task. In this work, I describe an innovative alternative to attacking the problem with hundreds of man-years of knowledge engineering. I propose to tap untrained volunteer contributors. The motivation for the contributors may range from the desire to win a prize or “exemplary contributor” status to the desire to help science to curiosity of seeing how their contribution makes the system better at estimating similarities and posing further questions. The approach of turning to volunteer contributors for a variety of tasks has been stated by the Open Mind initiative [9, 29], and has been directed at acquisition of commonsense knowledge by the Open Mind Common Sense project, described in [26, 27] and used in this work to provide a “seed” knowledge repository.

The main advantage of this approach is the large pool of millions of potential contributors, each presumably equipped with millions of axioms of commonsense knowledge. The challenge presented by this approach is lack of training of the contributors in knowledge representation; their main method of exchanging knowledge is

via natural language. To exploit the opportunity of collecting knowledge from such contributors, I have implemented and deployed LEARNER, a system which to date has collected knowledge from about 3,400 contributors.

To use the contributors’ efforts productively, in LEARNER, knowledge is collected actively, by formulating and posing questions that contributors answer. This approach, when contrasted with allowing contributors to simply type the knowledge in, has the advantages of ensuring that there is no redundancy in the knowledge being collected, as well as allows acquisition to be directed to the cases which are most likely to yield useful information. Both points are discussed further in Section 4.

The rest of the paper is organized as follows. In Section 2, I present the the knowledge representation scheme used in LEARNER, the major modules comprising the system, and the way contributors interact with the system. In Section 3, I present cumulative analogy (the method for formulating new knowledge acquisition questions from existing knowledge) and illustrate its operation with a step-by-step example. In Sections 4 and 5, I overview some related work, summarizing the volume and kinds of knowledge collected. I conclude with in Section 6, which contains a discussion of features of cumulative analogy that make it particularly suitable method for operating over incomplete (growing) and noisy collections of knowledge.

2. OVERVIEW OF LEARNER

To allow volunteers to add knowledge without any special training in knowledge representation, LEARNER accepts input in the form of statements in English. A contributor starts by entering a topic about which knowledge is to be acquired. LEARNER then determines, based on collected content, what other topics this topic is similar to. It then uses cumulative analogy to generate and present new specific questions about this topic. A sample screenshot of questions about “newspaper” is shown in Figure 1. Here, the system has used the statements it has previously collected to determine that the most similar topics are “book,” “map,” “magazine,” and “bag.” These are shown in the line beginning with the words “Similar topics.” From these, properties are mapped onto “newspaper” and presented as questions. For example, the first question shown is “newspapers contain information?” The knowledge acquisition questions are presented to the contributors in batches of up to 20. Each question can be either edited or answered as is using one of the multiple-choice answers: “Yes,” “No,” “Some / Sometimes,” “Matter of Opinion,” and “Nonsensical Question.”

2.1 Form of accepted knowledge

LEARNER imposes no restrictions on the domain of textual knowledge it operates on. It does, however, impose

Learning about **NEWSPAPER**

Teach about:

Examples: [beach](#), [chocolate](#), [computer](#)

Similar topics: [book](#) [i] 7.38, [map](#) [i] 3.01, [magazine](#) [i] 2.95, [bag](#) [i] 2.73

<input type="text" value="newspapers contain information?"/>	<input type="button" value="--Select--"/>	[i] (sc 3.05)
<input type="text" value="all newspapers have pages?"/>	<input type="button" value="--Select--"/>	[i] (sc 3.05)
<input type="text" value="newspapers are for reading?"/>	<input type="button" value="--Select--"/>	[i] (sc 3.05)

Figure 1: Screenshot of acquiring knowledge about “newspaper.”

some restrictions on the form of the accepted statements. LEARNER is designed to handle only single-sentence syntactically valid statements, each of which is interpretable on its own, in isolation. Furthermore, LEARNER is designed to handle primarily general factual knowledge about *generic concepts* (classes of objects), rather than specific events or individual objects. Here are some examples of statements LEARNER is designed to handle (and has in fact gathered):

“swans are white”	“keys can unlock locks”
“a hammer is a tool”	“computers are
“a yacht has a sail”	made up of parts”

To guide acquisition in the direction of factual knowledge about classes of objects, LEARNER enforces some syntactic constraints that limit what statements can be added. These constraints require that an statement (i) be parsable, (ii) be declarative rather than interrogative or imperative, and (iii) not start with referential words such as “that,” “these,” “my” and so on.

2.2 Seed knowledge and architecture

LEARNER uses the collected content to pose new knowledge acquisition questions. As such, LEARNER needs a significant knowledge collection to prime its analogical reasoning. The initial collection of commonsense knowledge LEARNER relied on consists of a seed knowledge repository of approximately 47,000 admissible statements of no longer than 7 words extracted from the knowledge collected by a different knowledge acquisition effort, Open Mind Common Sense (OMCS) [26, 27]. The OMCS effort collects knowledge without leveraging previously collected statements. Rather, it invites contributors to provide free-form knowledge and fill in a variety of templates. It is described further in Section 4.

Architecture-wise, LEARNER contains the following major components: web interface, question-generation component, growing collection of acquired statements, and

lexical knowledge (provided by WordNet [4]). LEARNER also has a natural language processing module to facilitate construction of internal representations and a database module to store and rapidly retrieve individual statements. The natural language processing module relies on the Link Grammar Parser [28] for parsing statements. The object-oriented database module for storing the knowledge repository is provided by FramerD [7].

2.3 Internal representation abstracts from syntactic details

The overall approach of LEARNER is to treat a sentence in natural language as a statement about an object O having a property P . For example, “cats eat mice” states that “cats” “eat mice.” Most admitted sentences can be viewed as statements about the sentence’s syntactic *subject*. Sentences which have a syntactic subject (such as “cats eat mice”) are also viewed as statements about the sentence’s syntactic *object*: in this example, “mice” have the property that “cats eat” (them).

LEARNER uses an internal, canonical representation which abstracts from syntactic details while capturing (well enough for the purposes of the system) what the statement expresses. In doing this LEARNER reduces both objects and their properties to their canonical form. For example, the statement “cats eat mice” would become “cat eat mouse”. To formulate a knowledge acquisition question, LEARNER bypasses the canonical form and processes the surface form of a statement on which the question is based (producing human-readable natural language).

To canonicalize a sentence, first, multi-word entities such as “fire engine” are identified by comparing against WordNet’s collection of multi-word entities. Next, only the base forms of the tokens appearing in the sentence

is used (the base form for a noun is its singular form, and for a verb the base form is its infinitive form). Determiners and closed-class words (such as “of,” “with,” “on,” and so on) are dropped by the canonicalization altogether.

For efficiency of comparing canonical forms, multi-word properties are stored as sorted lists of words (with their parts of speech), without preserving the order. LEARNER also tracks the syntactic role of the statement’s object O . This assures that two sentences such as “an elephant pushes a cart” and “a cart pushes an elephant” do not map to the same internal representation.

When creating statements from sentences, LEARNER attempts to handle negation correctly. Syntactic analysis of the sentence looks for “not” or constructs such as “don’t” or “cannot” negating the meaning of the sentence.

To represent the recognized negation, each statement in the knowledge repository has one of two *truth values*: $Tv = 1$ or 0 . When referring to a specific object-property pair O_i and P_j , I will denote these as $Tv(O_i, P_j) = 1$ and $Tv(O_i, P_j) = 0$, and informally refer to statements in these two classes as “is true” and “is false” statements, respectively.

Thus, the sentence “mice do not eat cats” actually gives rise to two statements, one about the syntactic subject and the other about the syntactic object. The one about the syntactic subject (hereafter subject-statement) can be denoted as:

$$Tv(\text{mouse}_{noun}, \{\text{cat}_{noun}, \text{eat}_{verb}\}) = 0$$

This notation does not distinguish between subject- and object-statements, and the distinction will not be important for the following discussion unless explicitly specified.

Because each statement consists of an object and a property, the entire knowledge repository can be visualized as a large matrix, with every known object of some statement being a row and every known property being a column. See Figure 2 for an illustration of an objects-properties matrix depicting a set of statements.

This approach to representing sentences as statements about objects having certain properties allows LEARNER to make use of simple statements with many syntactic variations. However, the approach has its limitations. A significant limitation is the system’s focus on “typically true” statements. The system is not equipped to reason about events which are true only in specialized situations, or otherwise require sophisticated delineation of context to be expressed. Another limitation of LEARNER is its exclusion of conjunctions or disjunctions from the set of statements it processes.

Objects	Properties (with simplified form)			
	contains knowledge ... contain knowledge	has pages have page	is cold be cold	is for reading be read
...
book	x	x		x
ice			x	
newspaper		x		x
magazine	x	x		x
...

Figure 2: Example of a matrix encoding statements about objects and their properties. The actual matrix has tens of thousands of rows and columns. For simplicity, only statements about a sentence’s subject are shown. “Is true” statements are marked by ‘x’es, and “is false” with ‘-’. Blank cells denote that truth values for these statements are not explicitly known.

The complex expressions (such as “houses have doors and windows”) are not handled by the system because of the difficulty of reformulating them in terms of more primitive statements automatically.

3. ALGORITHM AND SIMILARITY MEASURE

In this section, I present how reasoning by analogy from similar objects is used to formulate and pose new knowledge acquisition questions to the contributors. A novel feature of the analogical reasoning is its reliance on mapping properties from *many* similar objects, summing the evidence for posing any given question. Because evidence is accumulated from multiple objects, I call this method *cumulative analogy*.

The collection of knowledge starts with the contributor choosing a topic O_{tgt} about which knowledge will be acquired. Given O_{tgt} , cumulative analogy is performed in two stages: (i) **Select-NN**, selecting the set \mathcal{O} of nearest neighbors O_{src_i} based on similarity of these to O_{tgt} , and (ii) **Map-Props**, projecting known properties of \mathcal{O} not yet known about O_{tgt} onto O_{tgt} and presenting them as questions.¹ Both algorithms are explained on a simplified example of posing questions about “newspaper” in Sections 3.1 and 3.2.

3.1 Select-NN: Selecting the Nearest Neighbors

The steps of **Select-NN** are as follows. First, Select-NN retrieves the properties of the target object. Next, for each selected property, all statements about this property and some other object with this property are identified, regardless of whether this property “is true” or “is false” about that object.

Next, for every object that shares properties with “news-

¹It is the second step of projecting properties that motivates the terminology of “ O_{src} ” (“src” for “source”) and “ O_{tgt} ” (“tgt” for “target”).

paper,” the similarity between this object and “newspaper” is computed based on properties that they share and that they mismatch on (two objects mismatch on a property when a property is asserted as “is true” about one and as “is false” about the other). The detailed formulas used in computation of similarity are given below. Up to ten most similar objects are selected. These objects (in the example of “newspaper,” they would include “book” and “magazine”) together with their scores, are passed to the next stage of the algorithm, Map-Props.

The output of Select-NN — the source objects O_{src_i} paired with scored indicating their respective similarity to O_{tgt} — are passed to **Map-Props**, which proceeds as is described in the following section.

3.2 Map-Props: Mapping from similar objects to formulate questions

First, for every source object, Map-Props retrieves all properties asserted about this object. Both “is true” and “is false” statements are included. For the target object “newspaper,” the set of source objects may include “book” and “magazine.”

Next, each mentioned property is mapped back onto the target object, with the total score of the mapping depending on the similarity scores of the relevant source objects and the kind of property being mapped (subj or obj). Already known properties of the target object (“newspaper,” in our case) are filtered out, as are properties that “newspaper” can be expected to have by taxonomic reasoning (see below). Figure 3 illustrates the mapping of properties not yet known about “newspaper” back onto “newspaper.”

Objects	Properties (with simplified form)				
	contains knowledge	has pages	is cold	is for reading	...
...	contains knowledge	have page	be cold	be read	...
...
book	x	x		x	...
ice		-	x		...
newspaper	x?	x		x	...
magazine	x	x		x	...
...

Figure 3: Map-Props: Properties (such as “contains knowledge”) known about similar topics but not known about “newspaper” are formulated as questions about “newspaper” for presenting to contributors.

The specific formulas used in calculating the scores of the mapped properties are as follows. For every source object, the score of every property known about it is updated, according to the product of three weights:

$$TvWt(Tv(O, P)) \times PropClassWt(PropClass(P)) \times SimWt(SimSc(O))$$

The component weight functions are computed as fol-

lows:

$$TvWt(Tv) = \begin{cases} 1 & \text{if } Tv = 1, \\ -1 & \text{if } Tv = 0. \end{cases} \quad (1)$$

$$PropClassWt(PropClass) = \begin{cases} 1.1 & \text{if } PropClass \text{ is subj,} \\ 1 & \text{if } PropClass \text{ is obj.} \end{cases} \quad (2)$$

and

$$SimWt(SimSc) = 1 + \frac{\ln(SimSc)}{4}, \quad (3)$$

where $SimSc$ is the “similarity score” of O_{src} to O_{tgt} as returned by Select-NN. The equation for $TvWt$ simply ensures that the votes “for” and “against” this property holding for the target object are summed with the correct sign.

The equation for $SimWt$ (Eq. 3) ensures that the most similar objects in the set (according to Select-NN) have the greatest impact on the overall scores. A highly sub-linear function has been chosen to prevent overemphasizing any given object, and to produce a truly “cumulative” analogy.

Once the total scores for each property are computed, properties already asserted about O_{tgt} are eliminated, and up to 100 of the highest scoring remaining properties are returned for further filtering (at most 20 top scoring questions will be presented to the contributor). In practice, the computationally intensive work is the linguistic processing of a sentence that happens at later stages, so this over-generation does not cause a performance bottleneck. The properties with large negative scores (when such are present) are currently dropped, although they represent good hypothesis about what is *not* true about O_{tgt} .

3.3 Similarity measure is based on a theory of human similarity judgments

In selecting a similarity measure for use in Select-NN, many alternatives are possible. The similarity measure used in the current work is rooted in theory of human similarity judgments and captures some information-theoretic considerations deemed important in prior work on similarity (e.g. [24, 15]). While the current approach has performed well in the context of cumulative analogy, additional benefits may lie in incorporating automatic learning of ways to weight and combine features in selecting similar topics. A large body of work on adapting feature selection and learning weights for features can be found in the case-based reasoning (CBR) and machine learning traditions, e.g. [21, 10, 11, 32].

Empirical studies indicate that human similarity judgments tend to violate all of the properties of a distance metric typically expected of a similarity or distance *met-*

ric save perhaps for non-negativity. Symmetry, reflexivity [30] and triangle inequality [31, 6] can all be violated. These empirical results call into question the utility of using distance metrics such as the Minkowski metric or other proposed distance measures which would be unable to mimic these aspects of human similarity judgments [24, 15]. Instead, LEARNER’s similarity measure is derived from Tversky’s seminal contrast model of similarity. The model is formulated as:

$$\text{Sim}_{\text{Tversky}}(O, O') = \theta f(\mathcal{F}_O \cap \mathcal{F}_{O'}) - \alpha f(\mathcal{F}_O \setminus \mathcal{F}_{O'}) - \beta f(\mathcal{F}_{O'} \setminus \mathcal{F}_O)$$

where \mathcal{F}_O represents the set of features that an object O possesses, $\mathcal{F}_O \setminus \mathcal{F}_{O'}$ denotes features of O that O' does not have. The function f is typically assumed to be additive (simply returning the size of the set to which it is applied), and θ , α , and β are non-negative weights [30].

The exact formulas used in Select-NN are as follows. Let \mathcal{O}_P be the set of all objects O for which the property P has been asserted to be true (i.e., $\{O : Tv(O, P) = 1\}$), and $\|\mathcal{O}_P\|$ be the number of such objects. Then the *frequency weight* of P , denoted $\text{FreqWt}(P)$, can be defined as follows:

$$\text{FreqWt}(P) = \begin{cases} 2 & \text{if } \mathcal{O}_P = \emptyset, \\ 1 + 1/\log_2(\|\mathcal{O}_P\| + 1) & \text{otherwise.} \end{cases} \quad (4)$$

Note that the *FreqWt* ranges between 1 and 2, with larger values corresponding to properties that are true of fewer objects. The motivation for assigning lower weight to more common shared properties is that, other things being equal, two objects sharing a very rare property are probably more similar than two objects sharing a very common one. Giving greater weight to the rare features is consistent to prior approaches to measuring semantic similarity on information-theoretic grounds [24, 15], and reminiscent of inverse document frequency (IDF) term weighting methods used in information retrieval (see, e.g., [25]).

The amount by which the score of each object is updated is given by the function

$$\text{Wt}(Tv(O_{tgt}, P), Tv(O_{src}, P), P),$$

computed as follows:

$$\text{Wt}(Tv_1, Tv_2, P) = \begin{cases} \text{FreqWt}(P) & \text{if } (Tv_1, Tv_2) = (1, 1), \\ -1.5 & \text{if } (Tv_1, Tv_2) = (1, 0) \\ & \text{or } (0, 1), \\ 0 & \text{if } (Tv_1, Tv_2) = (0, 0). \end{cases} \quad (5)$$

Detailed rationale for these functions is beyond the scope of this paper; it can be found in [1].

3.4 Taxonomic reasoning augments similarity-based reasoning

The core algorithm for posing questions by cumulative analogy is simple in that it makes no assumptions about structure of knowledge. Using cumulative analogy alone, however, revealed that an additional mechanism, namely taxonomic reasoning, is needed to improve quality of questions. Without taxonomic reasoning, the algorithm will pose the same question about similar objects. For example, when told that “animals have body parts,” the generation component may, in generating questions for various topics, ask whether “monkeys have body parts,” “cats have body parts,” “dogs have body parts,” and so on.

In LEARNER, the generation of questions by analogy is augmented with a filter which relies on taxonomic reasoning. If a statement is a specialization of a (perhaps implicitly) universally quantified more general statement, it will not be presented as a question. Thus the specific questions will not be asked once the more general statement is known. The system does not attempt to generalize its knowledge, although that would constitute interesting future work.

Additionally, taxonomic information is used to filter out some similar topics in the output of Select-NN; this is a measure meant to remedy occasional generation of strange similarities when the system has insufficient knowledge. For example, if “oil” were judged to be similar to “mechanic,” “oil” would be filtered out on the grounds that, in WordNet there is not a path through the is-a taxonomy connecting “mechanic” and “oil”. This filtering does not apply to objects beyond WordNet’s dictionary.

4. RELATED WORK

The issue of collecting data from volunteer contributors has been studied in a different context by [9, 29], as part of Open Mind, a broader umbrella project concerned with collection of all kinds of data (handwriting samples, object segmentation, and so on) from volunteer contributors. In the area of collecting common-sense knowledge, the CYC team is investigating ways to use the CYC knowledge base to acquire from volunteers statements parsed into logic formulas with the volunteers’ confirmation.

Perhaps the closest in spirit project is Open Mind Common Sense (OMCS) [26, 27]. OMCS collects knowledge from contributors by offering a variety of knowledge entry activities, such as explaining the relationship between a pair of words or describing a picture. OMCS predates LEARNER and has served as an inspiration. Unlike LEARNER, however, OMCS does not leverage knowledge it has gathered to guide its future acquisition in any way. The OMCS contributors are able to browse the collected knowledge but do not see the system’s behavior improve or change as a result of the entered knowledge.

Any effort of collecting knowledge using natural language needs to have means of addressing ambiguity of words in natural language. The precision of questions formulated by LEARNER can benefit from disambiguation of word senses. Interestingly, such disambiguation can also be done by turning to human contributors, as is described in, e.g., [19, 3].

5. RESULTS

LEARNER does not require contributors to log in in order to contribute. This makes estimating the exact number of contributors difficult. It is possible, however, to calculate the total number of distinct IP addresses from which contributions have been made. Contributions have been made from a total of 3,427 IP addresses since the launch eleven months ago. The average contributing IP address has contributed 49.8 statements. Adding incentives and recognition of exemplary contributors may be a way to further increase the amount of contributions.

In total, 170,824 pieces of information have been added, by answering questions or entering new statements. Of these, 53.3% are labeled as “is true,” and 21.3% are labeled as “is false.” The number and percentage per answer type are shown in Table 1.

Answer	Num Entries	% of Total
Yes	91,066	53.3%
No	36,372	21.3%
Some/Sometimes	20,088	11.8%
Matter of Opinion	7,938	4.6%
Nonsensical Question	15,358	9.0%
Total:	170,824	100.0%

Table 1: Number of statements collected, by answer received.

A controlled experiment of a sample of 1,000 questions being posed using the seed knowledge repository indicated that when analogy is used to pose questions, 45% of questions were answered affirmatively. If, instead, Select-NN was replaced by a stub that instead used ten random objects from the knowledge repository as sources of similarity, only 8% of the questions were answered affirmatively by contributors. This experiment suggests that posing questions by analogy indeed results in more relevant questions.

Automatic template-based classifiers based on the parse structure of the sentence and appearance of certain words and expressions in it were implemented by hand to analyze the knowledge repository by the kind of statements contributed. According to these classifiers, 12.0% of the seed and 11.3% of the collected statements expressed taxonomic (is-a) knowledge; 8.0% of the seed and 9.7% of the collected statements expressed part-of relationships. The plurality of the statements (48.5% of the seed and 46.5% of the collected) expressed actions an

object tends to perform or a qualified action stating some qualification of how that action is typically performed. An example of an statement expressing an action is: “cats eat_{verb} mice.” An example of a qualified action is: “kangaroos jump_{verb} (very high)_{prep-phrase}.”

6. DISCUSSION

6.1 Usefulness of collected knowledge

The knowledge collected by LEARNER is similar in form to that collected by the Open Mind Common Sense (OMCS) project; the latter was in fact used to seed LEARNER’s knowledge acquisition. Thus, we can look at uses of OMCS knowledge (of which there have been several) to appraise potential applications of the knowledge collected by LEARNER. For example, collections of sentence-long natural language statements stating commonsense knowledge facts have been used for query reformulation in information retrieval [16], for making deductions about affect of text [17], and in inter-corpora word sense disambiguation experiments [2, 19].

In addition to these examples of potential uses, the knowledge gathered by LEARNER is already being actively used in LEARNER itself — the cumulative analogy algorithms use the collected knowledge to carry out the useful task of posing new knowledge acquisition questions.

6.2 Questions posed by cumulative analogy improve with acquisition of more knowledge

Two important properties of cumulative analogy, its *incremental payoff* and *noise tolerance* make it well suited to situations with partially unknown and ambiguous or unreliable knowledge. Cumulative analogy exhibits *incremental payoff* or *bootstrapping* qualities. That is, the more knowledge the system has, the more knowledge it can base its questions on. The replies to the knowledge acquisition questions formulated by analogy are immediately added to the knowledge repository, affecting the similarity calculations.

If Select-NN incorrectly rates a non-similar object as too similar, many knowledge acquisition questions posed with the contribution of this object are likely to be answered “no,” lowering its similarity and eventually causing it to be displaced by other, more similar objects.

Even answering questions affirmatively is likely to strengthen the similarity scores of topics that are more similar, while leaving scores of other topics in the set of similar topics unchanged. This process is also likely to improve the quality of future questions. Conversely, as more “is true” statements are added about the target object, new similar objects that also share those properties will tend to be uncovered leading to new analogies

and new KA questions being posed. Additionally, by summing evidence for and against each property, cumulative analogy focuses on the properties most likely to hold first.

6.3 Cumulative analogy is noise tolerant

Cumulative analogy is also *noise tolerant*. Noise tolerance is exhibited at two stages. Initially, Select-NN sums evidence for similarity from many individual properties, limiting the effects of spurious matches. Map-Props then sums evidence for each property from up to ten similar objects, further limiting the effect of any residual noise in Select-NN's output.

An example of cumulative analogy exhibiting noise tolerance in the step of creating a set of knowledge acquisition questions from the similar objects returned by Select-NN is as follows. Consider the similar objects to the object “**tool**” in the seed knowledge repository:

computer	7.61	car	3.90
machine	5.39	wrench	3.76
horseshoe	5.26	musical instrument	3.06
fire	3.92	fan	2.71
knife	3.90	weapon	2.45

Arguably, the similar topic “**fire**” (among others) is spurious. “**Tool**” was judged to be similar to “**fire**” because of the following pairs of statements: “Humans use tools/fire,” “A tool can help a person/Fire can help people,” and “Tools are useful/Fires can be useful.” Despite these matches, the top five questions about “**tools**” were as follows (shown together with the topics from which they were mapped):

tools can run on electricity? (computer, machine, fan)
 tools are machines? (computer, car, fan)
 a tool can hurt a person? (fire, car, weapon)
 tools are man-made? (computer, machine)
 tools are complicated? (computer, car)

In this case, only the third question was mapped with participation of “**fire**,” and this question had additional support from “**car**” and “**weapon**.” Other, irrelevant properties of “**fire**” present in the KB (e.g., “a fire is hot”, “fire consumes oxygen”, “fire can burn a house”) do not result in KA questions about “**tool**.”

Thus, cumulative analogy was able to focus in on the more relevant properties and similar objects, tolerating noise. Importantly, this noise can have many sources, making it a feature which addresses many problems. Noise sources can range from spurious matches that arise from insufficient knowledge (as in the above example) to lexical ambiguity to unintentional or malicious misinformation of the system by a contributor.

Additionally, the knowledge collected by LEARNER tends

to be *syntactically uniform*. By virtue of the algorithm (contributors are encouraged to confirm, deny, or correct statements mapped from other similar statements), syntactic variation is kept low, simplifying further processing and enabling the analogy mechanism to work well on consequent iterations.

7. FUTURE WORK

Some limitations of the current system are due to the simplicity of the internal representation over which the analogy operates. Interesting future work lies in exploring more advanced internal representation, for example allowing for “object, slot, value” expressions rather than “object, value” expressions or supporting variables could pose better knowledge acquisition questions and capture more nuanced statements. Also of great interest in augmenting the reasoning with a more detailed model of which features of objects imply presence or absence of which other features, going beyond correlation in both the reasoning and the collected knowledge.

Finally, the knowledge already collected represents a significant (although by no means complete) repository of world knowledge. Further applying this knowledge to address open problems in both commonsense reasoning and natural language processing also constitutes interesting and important future work.

In all, cumulative analogy is a tool which allows us to tackle an important problem of construction of commonsense knowledge repositories in an innovative way — by distributing the problem across volunteer contributors.

8. ACKNOWLEDGMENTS

Most of this work was carried out at the Massachusetts Institute of Technology Artificial Intelligence Laboratory, as part of PhD research under supervision of Patrick Winston. I gratefully acknowledge MIT, Patrick Winston, other members of thesis committee for helping develop this work, as well as Yolanda Gil and anonymous reviewers for extremely helpful comments.

9. REFERENCES

- [1] T. Chklovski. Using analogy to acquire commonsense knowledge from human contributors. Technical Report AITR-2003-002, Massachusetts Institute of Technology, February 2003. Available online at <ftp://publications.ai.mit.edu/ai-publications/2003/AITR-2003-002.pdf>.
- [2] T. Chklovski and R. Mihalcea. Building a sense tagged corpus with Open Mind Word Expert. In *Proceedings of the Workshop on "Word Sense Disambiguation: Recent Successes and Future Directions"*, pages 116–122, Philadelphia, PA, 2002. ACL. Available online at <http://citeseer.nj.nec.com/chklovski02building.html>.
- [3] T. Chklovski and R. Mihalcea. Exploiting agreement and disagreement of human annotators for word sense disambiguation. In *Proceedings of Recent Advances In NLP (RANLP 2003)*, September 2003.
- [4] C. Fellbaum, editor. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, Massachusetts, May 1998.

- [5] K. Forbus. Knowledge capture for bootstrapping intelligent systems. In Y. Gil, M. Musen, and J. Shavlik, editors, *Proceedings of the First International Conference on Knowledge Capture (K-CAP 2001)*. ACM, 2001.
- [6] R. Goldstone. *The MIT Encyclopedia of the Cognitive Sciences (MITECS)*, chapter Similarity, pages 763–765. MIT Press, Cambridge, MA, 1999. Available online at <http://cognitron.psych.indiana.edu/rgoldsto/pdfs/mitecs.pdf>.
- [7] K. Haase. FramerD: Representing knowledge in the large. *IBM Systems Journal*, 35(3&4):381–397, 1996.
- [8] P. Hayes. A catalog of temporal theories. Technical Report UIUC-BI-AI-96-01, University of Illinois, 1995.
- [9] M. Hearst, R. Hunson, and D. Stork. Building intelligent systems one e-citizen at a time. *IEEE Intelligent Systems [see also IEEE Expert]*, 14:16–20, May/June 1999.
- [10] J. Jarmulak, S. Craw, and R. Rowe. Self-optimising CBR retrieval. In *Proceedings of ICTAI-2000*, pages 376–383. IEEE Computer Society, 2000.
- [11] R. Kohavi, P. Langley, and Y. Yun. The utility of feature weighting in nearest-neighbor algorithms. In *Proceedings of the European Conference on Machine Learning (ECML97)*, 1997.
- [12] T. K. Landauer. How much do people remember? Some estimates of the quantity of learned information in long-term memory. *Cognitive Science*, 10(4):477–493, Oct–Dec 1986.
- [13] D. B. Lenat. CYC: A large-scale investment in knowledge infrastructure. *Communications of the ACM*, 38(11):33–38, 1995.
- [14] D. B. Lenat, R. V. Guha, K. Pittman, D. Pratt, and M. Shepherd. CYC: towards programs with common sense. *Communications of the ACM, (CACM), August 1990*, 33(8):30–49, 1990. Available online at <http://www.cyc.com/tech-reports/act-cyc-108-90/act-cyc-108-90.html>.
- [15] D. Lin. An information-theoretic definition of similarity. In *Proc. 15th International Conf. on Machine Learning*, pages 296–304. Morgan Kaufmann, San Francisco, CA, 1998. Available online at <http://citeseer.nj.nec.com/95071.html>.
- [16] H. Liu, H. Lieberman, and T. Selker. GOOSE: A goal-oriented search engine with commonsense. In *Proceedings of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web Based Systems (AH2002)*, Malaga, Spain, 2002.
- [17] H. Liu, H. Lieberman, and T. Selker. A model of textual affect sensing using real-world knowledge. In *Lexical Ambiguity Resolution*, pages 125–132, Miami, Florida, 2003.
- [18] J. McCarthy. Approximate objects and approximate theories. In A. G. Cohn, F. Giunchiglia, and B. Selman, editors, *Proceedings of the Conference on Principles of Knowledge Representation and Reasoning (KR-00)*, pages 519–526, S.F., Apr. 11–15 2000. Morgan Kaufman Publishers.
- [19] R. Mihalcea and T. Chklovski. Open mind word expert: Creating large annotated data collections with web users’ help. In *Proceedings of 4th International Workshop on Linguistically Interpreted Corpora (LINC-03) held in conjunction with EACL*, Budapest, April 2003.
- [20] M. Minsky. Commonsense-based interfaces. *Communications of the ACM*, 43(8):66–73, Aug. 2000.
- [21] D. Modha and S. Spangler. Feature weighting in K-means clustering. *Machine Learning*, 52(3), 2003.
- [22] D. I. Moldovan and R. Girju. An interactive tool for the rapid development of knowledge bases. *International Journal on Artificial Intelligence Tools*, 10(1-2):65–86, 2001. Available online at <http://citeseer.nj.nec.com/moldovan01interactive.html>.
- [23] S. G. Pauker, G. A. Gorry, J. P. Kassirer, and W. B. Schwartz. Toward the simulation of clinical cognition: Taking the present illness. *American Journal of Medicine*, 60:1–18, 1976. Available online at http://medg.lcs.mit.edu/people/psz/HST947_99/PIP-76-5.pdf.
- [24] P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI*, pages 448–453, 1995. Also available at <http://citeseer.nj.nec.com/resnik95using.html>.
- [25] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing & Management*, 24(5):513–523, 1988.
- [26] P. Singh. 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, 2002. AAAI. Available online at <http://citeseer.nj.nec.com/singh02public.html>.
- [27] P. Singh, T. Lin, E. Mueller, G. Lim, T. Perkins, and W. L. Zhu. Open Mind Common Sense: Knowledge acquisition from the general public. In *Proceedings of the First International Conference on Ontologies, Databases, and Applications of Semantics for Large Scale Information Systems*. Lecture Notes in Computer Science, Heidelberg: Springer-Verlag, 2002.
- [28] D. Sleator and D. Temperley. Parsing English with a link grammar. In *Third International Workshop on Parsing Technologies*, 1993. Available through: <ftp://bobo.link.cs.cmu.edu/pub/sleator/link-grammar>.
- [29] D. G. Stork and C. Lam. Open mind animals: Insuring the quality of data openly contributed over the world wide web. In *AAAI Workshop on learning with imbalanced data sets, American Association of Artificial Intelligence Meeting*, July 31 2000. Also available at <http://www.openmind.org/OpenMindAAAI.typeset.pdf>.
- [30] A. Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, July 1977.
- [31] A. Tversky and I. Gati. Similarity, separability and the triangle inequality. *Psychological Review*, 89(4):123–154, July 1982.
- [32] D. Wettschereck and D. W. Aha. Weighting features. In M. Veloso and A. Aamodt, editors, *Case-Based Reasoning, Research and Development, First International Conference*, pages pages 347–358, Berlin, 1995. Springer Verlag.