ConceptNet — a practical commonsense reasoning tool-kit

H Liu and P Singh

ConceptNet is a freely available commonsense knowledge base and natural-language-processing tool-kit which supports many practical textual-reasoning tasks over real-world documents including topic-gisting, analogy-making, and other context oriented inferences. The knowledge base is a semantic network presently consisting of over 1.6 million assertions of commonsense knowledge encompassing the spatial, physical, social, temporal, and psychological aspects of everyday life. ConceptNet is generated automatically from the 700 000 sentences of the Open Mind Common Sense Project — a World Wide Web based collaboration with over 14 000 authors.

1. Introduction

In today's digital age, text is the primary medium of representing and transmitting information, as evidenced by the pervasiveness of e-mails, instant messages, documents, weblogs, news articles, homepages, and printed materials. Our lives are now saturated with textual information, and there is an increasing urgency to develop technology to help us manage and make sense of the resulting information overload. While keyword-based and statistical approaches have enjoyed some success in assisting information retrieval, data mining, and natural language processing (NLP) systems, there is a growing recognition that such approaches deliver too shallow an understanding. To continue to make progress in textual-information management, vast amounts of semantic knowledge are needed to give our software the capacity for deeper and more meaningful understanding of text.

1.1 What is commonsense knowledge?

Of the different sorts of semantic knowledge that are researched, arguably the most general and widely applicable kind is knowledge about the everyday world that is possessed by all people — what is widely called 'commonsense knowledge'. While to the average person the term 'commonsense' is regarded as synonymous with 'good judgement', to the AI community it is used in a technical sense to refer to the millions of basic facts and understandings possessed by most people.

A lemon is sour. To open a door, you must usually first turn the doorknob. If you forget someone's birthday, they may be unhappy with you. Commonsense knowledge, thus defined, spans a huge portion of human experience, encompassing knowledge about the spatial, physical, social, temporal, and psychological aspects of typical everyday life. Because it is assumed that every person possesses commonsense, such knowledge is typically omitted from social communications, such as text. A full understanding of any text then, requires a surprising amount of commonsense, which currently only people possess. It is our purpose to find ways to provide such commonsense to machines.

1.2 Making sense of text

Since computers do not possess commonsense knowledge, it is understandable why they would be so bad at making sense of textual information. A computer can play chess quite well, yet it cannot even understand a simple children's story. A statistical classifier can categorise an e-mail as a 'flame', yet cannot explain why the author is incensed (most statistical classifiers use high-dimensional vector features which are nonsensical to a layperson). Given the sentence, 'I ate some chips with my lunch', a commonsense-deprived natural language understanding system is not likely to know that 'chips' probably refer to 'potato chips', and probably not 'computer chips'.

While keyword-spotting, syntactic language parsing, and statistical methods have all assisted in textual analysis, there is little substitute for the comprehensiveness and robustness of interpretation afforded by large-scale commonsense. Without commonsense, a computer reader might be able to guess that the sentence 'I had an awful day' is negative by spotting the mood keyword 'awful', but given the sentence 'I got fired today', the computer reader would not know what to think.

In contrast, a commonsense knowledge base should be able to reason about the situation of a person 'getting fired'. Perhaps it knows some things about 'getting fired'; people sometimes get fired because they are incompetent. A possible consequence of getting fired is not having money. People need money to pay for food and shelter. Even if the knowledge base does not have direct affective knowledge about 'getting fired', through its network of related knowledge it should be able to sense that the situation 'getting fired' usually bears many negative connotations such as fear, anger, and sadness.

commomsense knowledge spans a huge portion of human experience, but is typically omitted from social communications

Of course, commonsense knowledge is defeasible, meaning that it is often just a default assumption about the typical case (people might feel happy to be fired from a job they dislike); nevertheless, this sort of acontextual knowledge lays a critical foundation without which more nuanced interpretation cannot exist.

1.3 Introducing ConceptNet

Having motivated the significance of large-scale commonsense knowledge bases to textual information management, we introduce ConceptNet, a freely available large-scale commonsense knowledge base with an integrated natural-language-processing tool-kit that supports many practical textual-reasoning tasks over real-world documents.

The size and scope of ConceptNet make it comparable to, what are in our opinion, the two other most notable large-scale semantic knowledge bases in the literature: Cyc and WordNet. However, there are key differences, and these will be spelled out in the following section. While WordNet is optimised for lexical categorisation and word-similarity determination, and Cyc is optimised for formalised logical reasoning, ConceptNet is optimised for making practical context-based inferences over real-world texts. That it reasons simply and gracefully over text is perhaps owed to the fact that its knowledge representation is itself semi-structured English (a further discussion of reasoning in natural language can be found in Liu and Singh [1]).

ConceptNet is also unique from Cyc and WordNet for its dedication to contextual reasoning. Of the 1.6 million assertions in its knowledge base, approximately 1.25 million are dedicated to different sorts of generic conceptual connections called k-lines (a term introduced by Minsky [2]). Contextual commonsense reasoning, we argue, is highly applicable to textual information management because it allows a computer to broadly characterise texts along interesting dimensions such as topic and affect; it also allows a computer to understand novel or unknown concepts by employing structural analogies to situate them within what is already known.

By integrating the ConceptNet knowledge base with a natural-language-processing engine, we dramatically reduce the engineering overhead required to leverage common sense in applications, obviating the need for specialised expertise in commonsense reasoning or natural language processing. ConceptNet has, in its two years of existence, been used to drive tens of interesting applications, many of which were engineered by MIT undergraduate and graduate students within the timeframe of a school semester.

We believe that the ConceptNet tool-kit represents a new direction for the development of commonsense AI systems. By making many previously inaccessible technical feats possible and even simple to engineer, ConceptNet enables a new commonsense AI research agenda, grounded not in toy systems for esoteric domains, but in novel real-world applications that provide great value to everyone.

1.4 Paper's organisation

The rest of this paper is organised as follows. Firstly, we give a detailed comparison of our approach to those of Cyc and WordNet. Secondly, we present a brief history of ConceptNet and describe how it was built, and how it is structured. Thirdly, ConceptNet's integrated natural-language-processing engine is presented along with a review of the various contextual reasoning tasks that it supports. Fourthly, we present a technical quantitative and qualitative analysis of the ConceptNet knowledge base and tool-kit. Fifthly, we briefly review the many research applications that have been developed using ConceptNet. We conclude with further reflection on how ConceptNet fits into a bigger picture.

2. ConceptNet, Cyc, and WordNet

In our introductory remarks, we motivated the need for a commonsense knowledge base; however, the task of assembling together such a thing is far from trivial. Representing and amassing large-scale commonsense has been an elusive dream since the conception of artificial intelligence some fifty years ago.

It has historically been quite daunting because of the sheer breadth and size of knowledge that must be amassed, and the lack of certainty in how the knowledge is best represented. A founder of AI, Marvin Minsky, once estimated that '... commonsense is knowing maybe 30 or 60 million things about the world and having them represented so that when something happens, you can make analogies with others' [3].

In our opinion, the literature's two most notable efforts to build large-scale, general-purpose semantic knowledge bases are WordNet and Cyc.

Begun in 1985 at Princeton University, WordNet [4] is arguably the most popular and widely used semantic resource in the computational linguistics community today. It is a database of words, primarily nouns, verbs and adjectives, organised into discrete 'senses', and linked by a small set of semantic relations such as the synonym relation and 'is-a'

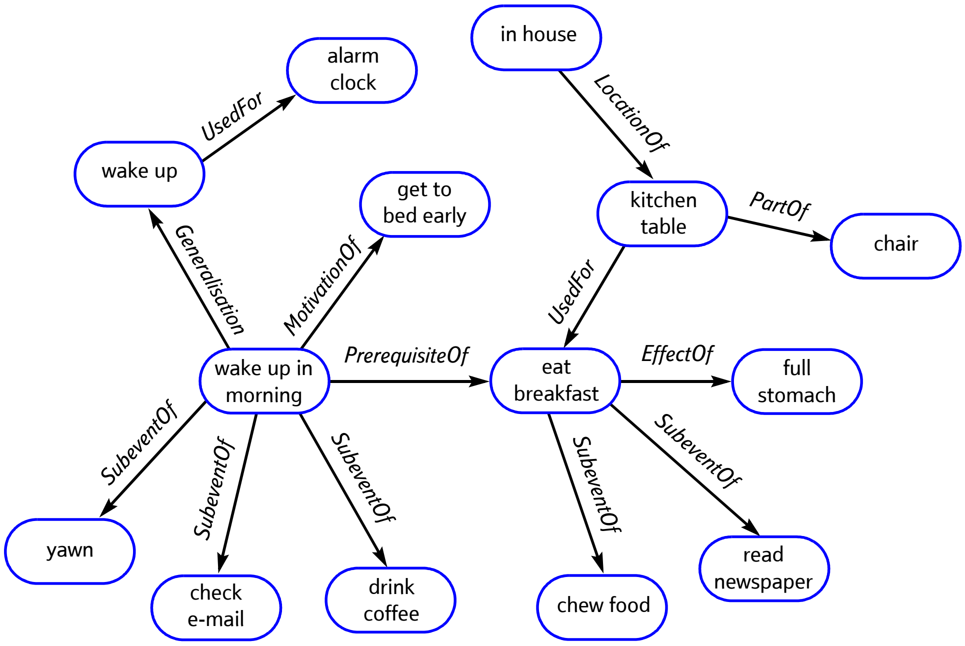

Fig 1 An excerpt from ConceptNet's semantic network of commonsense knowledge. Compound (as opposed to simple) concepts are represented in semi-structured English by composing a verb (e.g. 'drink') with a noun phrase ('coffee') or a prepositional phrase ('in morning').

hierarchical relations. Its most recent version 2.0 contains roughly 200 000 word 'senses' (a sense is a 'distinct' meaning that a word can assume). One of the reasons for its success and wide adoption is its ease of use. As a simple semantic network with words at the nodes, it can be readily applied to any textual input for query expansion, or determining semantic similarity. ConceptNet also adopts a simple-to-use semantic network knowledge representation, but rather than focusing on formal taxonomies of words, ConceptNet focuses on a richer (though still very pragmatic) set of semantic relations (e.g. EffectOf, DesireOf, CapableOf) between compound concepts (e.g. 'buy food', 'drive car').

rather than handcrafting commonsense knowledge, OMCS turned to the general public

The Cyc project, begun in 1984 by Doug Lenat, tries to formalise commonsense knowledge into a logical framework [5]. Assertions are largely handcrafted by knowledge engineers at Cycorp, and as of 2003, Cyc has over 1.6 million facts interrelating more than 118 000 concepts (source: cyc.com). To use Cyc to reason about text, it is necessary to first map the text into its proprietary logical representation, described by its own language CycL. However, this mapping process is quite complex because all of the inherent ambiguity in natural language must be resolved to produce the unambiguous logical formulation required by CycL. The difficulty of applying Cyc to practical textual reasoning tasks, and the present unavailability of its full content to the general public, make it a prohibitive option for most textual-understanding tasks.

By comparison, ConceptNet is a semantic network of commonsense knowledge that at present contains 1.6 million edges connecting more than 300 000 nodes. Nodes are semi-structured English fragments, interrelated by an ontology of twenty semantic relations. A partial snapshot of actual knowledge in ConceptNet is given in Fig 1. When examining the sizes of ConceptNet, WordNet, and Cyc, we would like to give the caveat that numbers provide at best a tenuous dimension of comparison. As ConceptNet, WordNet, and Cyc all employ different knowledge representations, cross-representational numeric comparisons may not be particularly meaningful.

2.1 Differences in acquisition

While WordNet and Cyc are both largely handcrafted by knowledge engineers, ConceptNet is generated automatically from the English sentences of the Open Mind Common Sense (OMCS) corpus. Rather than manually handcrafting commonsense knowledge, OMCS turns to the general public for help. The idea is that every lay person can contribute commonsense knowledge to our project because it is knowledge that even children possess. In 2000, one of the authors launched the Open Mind Common Sense Web site [6] as a World Wide Web based collaborative project. Thanks to the over 14 000 Web contributors who logged in to enter sentences in a fill-in-the-blank fashion (e.g. 'The effect of eating food is …'; 'A knife is used for …'), we amassed over 700 000 English sentences of commonsense. By applying natural language processing and extraction rules to the semi-structured OMCS sentences, 300 000 concepts and 1.6 million binary-relational assertions are extracted to form ConceptNet's semantic network knowledge base. While both the WordNet and Cyc projects have been amassing knowledge for about 20 years, the OMCS project has successfully employed Web collaboration to amass a great amount of commonsense knowledge in a relatively short time and at a tiny fraction of the cost.

2.2 Structured like WordNet, relationally rich like Cyc

ConceptNet can best be seen as a semantic resource that is structurally similar to WordNet, but whose scope of contents is general world knowledge in the same vein as Cyc. We have taken the simple WordNet framework and extended it in three principal ways.

Firstly, we extend WordNet's notion of a node in the semantic network from purely lexical items (words and simple phrases with atomic meaning) to include higher-order compound concepts, which compose an action verb with one or two direct or indirect arguments (e.g. 'buy food', 'drive to store'). This allows us to represent and author knowledge around a greater range of concepts found in everyday life, such as events (e.g. 'buy food', 'throw baseball', 'cook dinner'). On the flipside, because the corpus from which ConceptNet gets generated is not word-sense-tagged, ConceptNet does not currently distinguish between word senses. There is, however, an affiliated project called OMCSNet-WNLG [7] that is sense-disambiguating ConceptNet nodes.

Secondly, we extend WordNet's repertoire of semantic relations from the triplet of synonym, is-a, and part-of, to a present repertoire of twenty semantic relations including, for example, EffectOf (causality), SubeventOf (event hierarchy), CapableOf (agent's ability), PropertyOf, LocationOf, and MotivationOf (affect). Some further intuition for this relational ontology is given in the next section of the paper. Although ConceptNet increases the number and variety of semantic relations, engineering complexity is not necessarily increased. Many contextual reasoning applications of the ConceptNet semantic network either do not require any distinguishment of the relations, or at most require only coarse groupings of relations to be distinguished (e.g. affect-relations versus temporal-relations versus spatial-relations). Furthermore, the complexities of the relational ontology are largely taken care of by the ConceptNet textual reasoning tool-kit. By automating many kinds of interesting inference, the tool-kit can drastically reduce complexity involved in engineering common sense into applications.

Thirdly, when compared to WordNet, the knowledge in ConceptNet is of a more informal, defeasible, and practically valued nature. For example, WordNet has formal taxonomic knowledge that 'dog' is a 'canine', which is a 'carnivore', which is a 'placental mammal'; but it cannot make the practically oriented member-to-set association that 'dog' is a 'pet'. Unlike WordNet, ConceptNet also contains a lot of knowledge that is defeasible, meaning it describes something that is often true, but not always, e.g. EffectOf('fall off bicycle', 'get hurt'). A great deal of our everyday world knowledge is defeasible in nature, and we cannot live without it.

2.3 ConceptNet as a context machine

While ConceptNet, WordNet, and Cyc all purport to capture general-purpose world-semantic knowledge, the qualitative differences in their knowledge representations make them suitable for very different purposes. Because WordNet has a lexical emphasis and largely employs a formal taxonomic approach to relating words (e.g. 'dog' is-a 'canine' is-a 'carnivore' is-a 'placental mammal'), it is most suitable for lexical categorisation and word-similarity determination.

Because Cyc represents commonsense in a formalised logical framework, it excels in careful deductive reasoning and is appropriate for situations which can be posed precisely and unambiguously.

ConceptNet, in contrast, excels at contextual commonsense reasoning over real-world texts. In his treatise critiquing the traditional AI dogma on reasoning, AI researcher Gelernter [8] characterises human reasoning as falling along a spectrum of mental focus. When mental focus is high, logical and rational thinking happens. Traditional AI only baptises this extremity of the spectrum as being 'reasoning'. However, Gelernter is quick to point out that much, if not the vast majority, of human reasoning happens at a medium or low focus, where crisp deduction is traded in for gestalt perception, creative analogy, and at the lowest focus, pure association. Even if we are skeptical of Gelernter's folk psychology, the importance of contextual reasoning is hard to deny. Without understanding the gestalt context behind a sentence or a story, we would not be able to prefer certain interpretations of ambiguous words and descriptions to others. Without a context of expectations to violate, we would not be able to understand many examples of sarcasm, irony, or hyperbole. Without weaving story-bits together into a contextual fabric, we would not be able to skim a book and would have to read it word by word. Just as people need this sort of contextual mechanism to read, computer readers will likewise require contextual reasoning to intelligently manage textual information. If computers could be taught to be better contextual reasoners, it would revolutionise textual information management. We believe that ConceptNet is making progress towards this goal.

ConceptNet invests in making associations, even ones whose value is not immediately apparent

Like WordNet, ConceptNet's semantic network is amenable to context-friendly reasoning methods such as spreading activation [9] (think — activation radiating outward from an origin node) and graph traversal. However, since ConceptNet's nodes and relational ontology are more richly descriptive of everyday commonsense than WordNet's, better contextual commonsense inferences can be achieved, and require only simple improvements to spreading activation. Context-based inference methods allow ConceptNet to perform interesting tasks such as the following:

- 'given a story describing a series of everyday events, where is it likely that these events will take place, what is the mood of the story, and what are possible next events?' (spatial, affective, and temporal projections),

- ‘given a search query (assuming the terms are commonsensical) where one of the terms can have multiple meanings, which meaning is most likely?’ (contextual disambiguation),
- ‘presented with a novel concept appearing in a story, which known concepts most closely resemble or approximate the novel concept?’ (analogy-making).

Two key reasons why ConceptNet is adept at context are its investment in associational knowledge, and its natural language knowledge representation. More than WordNet and more than Cyc, ConceptNet invests heavily in making associations between concepts, even ones whose value is not immediately apparent. Of the 1.6 million facts interrelating the concepts in the ConceptNet semantic network, approximately 1.25 million are dedicated to making rather generic connections between concepts. This type of knowledge is best described as k-lines, which Minsky [2] implicates as a primary mechanism for context and memory. ConceptNet’s k-line knowledge increases the connectivity of the semantic network, and makes it more likely that concepts parsed out of a text document can be mapped into ConceptNet.

ConceptNet’s natural language knowledge representation also benefits contextual reasoning. Unlike logical symbols, which have no a priori meaning, words are always situated in connotations and possible meanings. That words carry prior meanings, however, is not a bad thing at all, especially in the context game. By posing ConceptNet’s nodes as semi-structured English phrases, it is possible to exploit lexical hierarchies like WordNet to make node-meanings flexible. For example, the nodes ‘buy food’ and ‘purchase groceries’ can be reconciled by recognising that ‘buy’ and ‘purchase’ are in some sense synonymous, and that ‘groceries’ are an instance of ‘food’.

A criticism that is often levied against natural language knowledge representations is that there are many ambiguous and redundant ways to specify the same idea. We maintain that these ‘redundant’ concepts can be reconciled through background linguistic knowledge if necessary, but there is also value to maintaining different ways of conveying the same idea (e.g. ‘car’ and ‘automobile’ are almost the same, but may imply different contextual nuances, such as formality of discourse). On the subject of ConceptNet’s natural language knowledge representation, we have dedicated an entire other paper [1].

In summary, we have discussed the relationship between ConceptNet and the two most notable predecessor projects WordNet and Cyc. Whereas Cyc and WordNet are largely handcrafted resources each built over a project lifetime of 20 years, ConceptNet is automatically built by extraction from the sentences of the Open Mind Common Sense project, a corpus built over the past four years by 14 000 Web collaborators. ConceptNet embraces the ease-of-use of WordNet’s semantic network representation, and the richness of Cyc’s content. While WordNet excels as a lexical resource, and Cyc excels at unambiguous logical deduction, ConceptNet’s forte is contextual commonsense reasoning — making practical inferences over real-world texts, such as analogy, spatial-temporal-affective projection, and contextual disambiguation. We believe that the innovation of contextual reasoning about texts can inspire major rethinking of what is possible in textual information management.

In the next section, we take a retrospective look at the origins of ConceptNet, and then we describe how the knowledgebase is built and structured.

3. Origin, construction and structure of ConceptNet

In this section, we first explain the origins of ConceptNet in the Open Mind Common Sense corpus; then we demonstrate how knowledge is extracted to produce ConceptNet’s semantic network; and finally, we describe the structure and semantic content of the network. Version 2.0 of the ConceptNet knowledge base, knowledge browser program, and the integrated natural-language-processing tool-kit are available for download at www.conceptnet.org.

3.1 History of ConceptNet

Until recently, it seemed that the only way to build a commonsense knowledge base was through the expensive process of hiring an army of knowledge engineers to hand-code each and every fact à la Cyc. However, inspired by the success of distributed and collaborative projects on the Web, we turned to volunteers from the general public to massively distribute the problem of building a commonsense knowledge base. In 2000, the Open Mind Common Sense (OMCS) Web site [6] was built, a collection of 30 different activities, each of which elicits a different type of commonsense knowledge-simple assertions, descriptions of typical situations, stories describing ordinary activities and actions, and so forth. Since then the Web site has gathered over 700 000 sentences of commonsense knowledge from over 14 000 contributors from around the world, many with no special training in computer science. The OMCS corpus now consists of a tremendous range of different types of commonsense knowledge, expressed in natural language. The OMCS sentences alone, however, are not directly computable.

> ConceptNet’s forte is making practical inferences across real-world texts

The earliest application of the OMCS corpus to a task made use of the OMCS sentences by employing extraction rules to mine out knowledge into a semantic network. The ARIA photo retrieval system’s commonsense robust inference system (CRIS) [10] had the idea to extract taxonomic, spatial, functional, causal, and emotional knowledge from OMCS, populate a semantic network, and use spreading activation to improve information retrieval. CRIS, then, was the earliest precursor to ConceptNet, which has undergone several generations of re-invention.

The innovation of CRIS to information retrieval suggested a new approach to building a commonsense knowledge base. Rather than directly engineering the knowledge structures used by the reasoning system, as is done in Cyc, OMCS encourages people to provide information clearly in natural language. From these semi-structured English sentences, we are able to extract out knowledge into more computable representations. Elaborating on CRIS, we built a semantic network called OMCSNet by systematically reformulating all the semi-structured sentences of OMCS into a semantic network with 280 000 edges and 80 000 nodes. We also developed an API for OMCSNet, supporting three chief functions — FindPathsBetween Nodes(node1,node2), GetContext(node), and Get AnalogousConcepts(node). The OMCSNet package was used by early adopters to build several interesting applications, such as a dynamically generated foreign-language phrasebook called GloBuddy (a newer version is discussed by Lieberman et al [11]), and a conversational topic spotter [12].

Furthermore, OMCSNet was widely adopted by undergraduate and masters-level students seeking to do term projects for an MIT Media Lab seminar called Common Sense Reasoning for Interactive Applications (taught by Henry Lieberman in 2002 and 2003). Using OMCSNet, these students were able to engineer a diverse collection of interesting applications ranging from an AI-version of the game, Taboo, to a financial commonsense advisor, to an automatically generated gaming environment [13]. It was promising to see that within the window of a school semester, applications such as these could be engineered. From these early adopters, we also observed that the integration of natural language processing and OMCSNet remained an engineering hurdle, and we wanted to address this issue in our next iteration of the tool-kit.

3.2 ConceptNet 2.0

ConceptNet is the latest incarnation of CRIS/OMCSNet. It is the primary machine-computable form of the Open Mind Common Sense corpus. The current version 2.0 features 1.6 million assertions interrelating 300 000 nodes. A new system for weighting knowledge is implemented, which scores each binary assertion based on how many times it was uttered in the OMCS corpus, and on how well it can be inferred indirectly from other facts in ConceptNet. Syntactic and semantic constraints were added to the extraction rules mapping OMCS sentences to ConceptNet assertions; in particular, we wanted to enforce a syntactic/semantic grammar on the nodes, in order to improve the normalisation process.

Multiple assertions are now inferred from a single Open Mind sentence. For example, from the sentence, ‘A lime is a sour fruit’, we extract the knowledge, IsA(lime, fruit) but additionally infer PropertyOf(lime, sour). Generalisations are also inferred. For example, if the majority of fruits have the property ‘sweet’, then this property is lifted to the parent class, as: Property Of(fruit, sweet).

Three k-line relations (SuperThematicKLine, ThematicKLine, and ConceptuallyRelatedTo) were also mined from the OMCS corpus and added as a feature in ConceptNet. This is motivated by an increasing recognition by the authors of the value of ConceptNet to problems of context. SuperThematicKLines, which unify themes with their variations (e.g. ‘buy’ is a supertheme of ‘purchase groceries’ and ‘buy food’), are also steps towards achieving new flexibility for nodes, allowing advanced manipulations such as node reconciliation (e.g. dynamically merge ‘buy food’ and ‘purchase groceries’ given the appropriate context) and node-variation generation (i.e. applying lexical hierarchies and synonyms to generate similar nodes). This should help ConceptNet to better map to surface linguistic variations present in real-world texts.

Perhaps the most compelling new feature in ConceptNet version 2.0 is the integration of the MontyLingua natural-language-processing engine [14]. MontyLingua is an end-to-end integrated natural-language-understander for English written in Python and also available in Java. Whereas earlier ConceptNet APIs only accepted the input of well-normalised English phrases, the new API accepts the input of paragraphs and documents, automatically extracts salient event-structures from parsed text, and performs the requested inferences using the semantic network. The types of inferencing tasks currently supported are discussed in a later section. We think of MontyLingua as a key integration because it eliminates familiarity with natural language processing as a major engineering hurdle to the adoption of commonsense reasoning for many textual-information management applications.

3.3 Building ConceptNet

ConceptNet is produced by an automatic process, which first applies a set of extraction rules to the semi-structured English sentences of the OMCS corpus, and then applies an additional set of ‘relaxation’ procedures (i.e. filling in and smoothing over network gaps) to optimise the connectivity of the semantic network.

3.3.1 Extraction phase

Approximately fifty extraction rules are used to map from OMCS’s English sentences into ConceptNet’s binary-relation assertions. This is facilitated by the fact that the OMCS Web site already elicits knowledge in a semi-structured way by prompting users with fill-in-the-blank templates (e.g. ‘The effect of [falling off a bike] is [you get hurt]’). Sentences for which there are no suitable relation-types may still be extracted into the generic, ‘ConceptuallyRelatedTo’ k-line relation if they contain semantically fruitful terms. Extraction rules are regular expression patterns crafted to exploit the already semi-structured nature of most of the OMCS sentences. In addition, each sentence is given a surface parse by MontyLingua so that syntactic and semantic constraints can be enforced on the nodes.

As a result, nodes in ConceptNet have guaranteed syntactic structure, facilitating their computability. Each node is an English fragment composed out of combinations of four syntactic constructions — verbs (e.g. ‘buy’, ‘not eat’, ‘drive’), noun phrases (e.g. ‘red car’, ‘laptop computer’), prepositional phrases (e.g. ‘in restaurant’, ‘at work’), and adjectival phrases (e.g. ‘very sour’, ‘red’). Their order is also restricted such that verbs must precede noun phrases and adjectival phrases, which in turn must precede prepositional phrases.

3.3.2 Normalisation phase

Extracted nodes are also normalised. Errant spelling is corrected by an unsupervised spellchecker, and syntactic constructs (i.e. verbs, noun phrases, prepositional phrases, and adjectival phrases) are stripped of determiners (e.g. 'the' and 'a'), modals, and other semantically peripheral features. Words are stripped of tense (e.g. 'is/are/were'→'be') and number (e.g. 'apples'→'apple'), reducing them to a canonical 'lemma' form.

3.3.3 Relaxation phase

After the extraction phase produces a list of normalised assertions, a further level of processing performs 'relaxation' over the network, meant to smooth over semantic gaps and to improve the connectivity of the network. Firstly, duplicate assertions are merged (since many common facts are uttered multiple times) and an additional metadata field called 'frequency' is added to each predicate-relation to track how many times something is uttered. Secondly, the 'IsA' hierarchical relation is used to heuristically 'lift' knowledge from the children nodes to the parent node. An example of this is given below:

```
[(IsA 'apple' 'fruit');
 (IsA 'banana' 'fruit');
 (IsA 'peach' 'fruit')]

AND

[(PropertyOf 'apple' 'sweet');
 (PropertyOf 'banana' 'sweet');
 (PropertyOf 'peach' 'sweet')]

IMPLIES

 (PropertyOf 'fruit' 'sweet')
```

Thirdly, thematic and lexical generalisations are produced which relate more specific knowledge to more general knowledge, and these fall under the SuperThematicKLine relation-type. WordNet and FrameNet's [15] verb synonym-sets and class-hierarchies are used. Two examples of these generalisations are given below:

```
(SuperThematicKLine 'buy food' 'buy')
(SuperThematicKLine 'purchase food' 'buy')
```

Fourthly, when noun phrase nodes contain adjectival modifiers, these can be 'lifted' and reified as additional PropertyOf knowledge, as given in the following example:

```
[(IsA 'apple' 'red round object');
 (IsA 'apple' 'red fruit')]

IMPLIES

(PropertyOf 'apple' 'red')
```

Fifthly, vocabulary discrepancies and morphological variations are reconciled. Vocabulary differences like 'bike' and 'bicycle' are bridged. Morphological variations such as 'relax'/'relaxation', (action versus state) or 'sad'/'sadness' (adjective/nominal) are also reconciled by the addition of a lexical SuperThematicKLine.

To track knowledge generated by these additional generalisations, a metadata field called 'inferred_frequency' is added to each predicate-relation. As we shall see later in this paper, the ConceptNet tool-kit's inference procedures treat inferred-knowledge as inferior to uttered-knowledge, but nonetheless use them at a discount. Although all the additional knowledge extracted from this relaxation phase could theoretically be performed at the runtime of inference, inferring them at build-time saves much computational expense associated with the use of natural-language-processing techniques.

3.4 Structure of the ConceptNet knowledge base

The ConceptNet knowledge base is formed by the linking together of 1.6 million assertions (1.25 million of which are k-lines) into a semantic network of over 300 000 nodes. The present relational ontology consists of twenty relation-types. Figure 2 is a treemap of the ConceptNet relational ontology, showing the relative amounts of knowledge falling under each relation-type. Table 1 gives a concrete example of each relation-type.

Table 1 ConceptNet's twenty relation-types are illustrated by examples from actual ConceptNet data. The relation-types are grouped into various thematics. *f* counts the number of times a fact is uttered in the OMCS corpus. *i* counts how many times an assertion was inferred during the 'relaxation' phase.

K-LINES (1.25 million assertions) (ConceptuallyRelatedTo 'bad breath' 'mint' '*f*=4;*i*=0;') (ThematicKLine 'wedding dress' 'veil' '*f*=9;*i*=0;') (SuperThematicKLine 'western civilisation' 'civilisation' '*f*=0;*i*=12;')
THINGS (52 000 assertions) (IsA 'horse' 'mammal' '*f*=17;*i*=3;') (PropertyOf 'fire' 'dangerous' '*f*=17;*i*=1;') (PartOf 'butterfly' 'wing' '*f*=5;*i*=1;') (MadeOf 'bacon' 'pig' '*f*=3;*i*=0;') (DefinedAs 'meat' 'flesh of animal' '*f*=2;*i*=1;')
AGENTS (104 000 assertions) (CapableOf 'dentist' 'pull tooth' '*f*=4;*i*=0;')
EVENTS (38 000 assertions) (PrerequisiteEventOf 'read letter' 'open envelope' '*f*=2;*i*=0;') (FirstSubeventOf 'start fire' 'light match' '*f*=2;*i*=3;') (SubeventOf 'play sport' 'score goal' '*f*=2;*i*=0;') (LastSubeventOf 'attend classical concert' 'applaud' '*f*=2;*i*=1;')
SPATIAL (36 000 assertions) (LocationOf 'army' 'in war' '*f*=3;*i*=0;')
CAUSAL (17 000 assertions) (EffectOf 'view video' 'entertainment' '*f*=2;*i*=0;') (DesirousEffectOf 'sweat' 'take shower' '*f*=3;*i*=1;')
FUNCTIONAL (115 000 assertions) (UsedFor 'fireplace' 'burn wood' '*f*=1;*i*=2;') (CapableOfReceivingAction 'drink' 'serve' '*f*=0;*i*=14;')
AFFECTIVE (34 000 assertions) (MotivationOf 'play game' 'compete' '*f*=3;*i*=0;') (DesireOf 'person' 'not be depressed' '*f*=2;*i*=0;')

ConceptNet's relational ontology was determined quite organically. The original OMCS corpus was built largely through its users filling in the blanks of templates like 'a hammer is for ...'. Other portions of the OMCS corpus accepted freeform input, but restricted the length of the input so as to encourage pithy phrasing and simple syntax. ConceptNet's choice of relation-types reflect our original choice of templates in OMCS, and also reflect common patterns we observed in the freeform portion of the corpus.

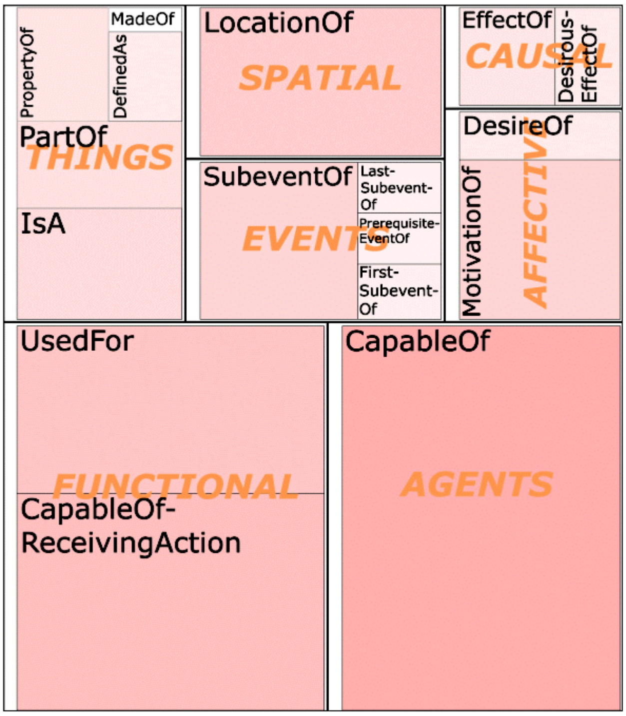

Fig 2 A treemap of ConceptNet's relational ontology (with the three k-line relations omitted). Relation types are grouped into various thematics and the relative sizes of the rectangles are proportional to the number of assertions belonging to each relation-type.

In summary, ConceptNet is the primary machine-computable resource offered by the Open Mind Common Sense project. First built in 2002, it has since undergone several generations of revision motivated by feedback from early adopters of the system. The present ConceptNet version 2.0 consists of both a semantic network, and an integrated natural-language-processing tool-kit (MontyLingua [14]). The ConceptNet knowledge base is built by an automated three-stage process:

- regular expressions and syntactic-semantic constraints extract binary-relation assertions from OMCS sentences,
- assertions are normalised,
- heuristic 'relaxation' over the assertion-base produces additional 'intermediate' knowledge such as semantic and lexical generalisations, which helps to bridge other knowledge and to improve the connectivity of the knowledge base.

The ConceptNet knowledge base consists of 1.25 million k-line assertions and 400 000 non-k-line assertions, distributed into twenty organically decided relation-types.

Having characterised ConceptNet's origin, construction, and structure, we now discuss how the knowledge base is leveraged by the tool-kit to address various textual-reasoning tasks.

4. Practical commonsense reasoning with the ConceptNet tool-kit

Whereas logic is microscopic, highly granular, well-defined, and static, context is macroscopic, gestalt, heuristic, and quite dynamic. ConceptNet excels at problems of context because it is more invested in the many ways that commonsense concepts relate to one another, rather than obsessing over the truth conditions of particular assertions. By nuancing network-based reasoning methods such as spreading activation to take advantage of ConceptNet's relational-ontology, various contextual-commonsense-reasoning tasks can be achieved.

In this section, we firstly present ConceptNet's integrated natural-language-processing engine. Secondly, we discuss three basic node-level reasoning capabilities persisting from previous versions of ConceptNet — contextual neighbourhoods, analogy and projection. Thirdly, we present four document-level reasoning capabilities newly supported in ConceptNet — topic gisting, disambiguation/classification, novel-concept identification, and affect sensing.

4.1 An integrated natural-language-processing engine

ConceptNet version 2.0's integrated natural-language-processing engine is an adapted version of the MontyLingua natural-language understander [14]. MontyLingua is written in cross-platform Python, but is also available as a Java library, or the whole ConceptNet package can be run as an XML-RPC server (included with the distribution) and accessed via sockets.

ConceptNet is produced from the structured English of OMCS by an automated process

MontyLingua performs language-processing functions including text normalisation, commonsense-informed part-of-speech tagging, semantic recognition, chunking, surface parsing, lemmatisation, thematic-role extraction, and pronominal resolution. The simplest evocation of MontyLingua takes as input a raw text document and outputs a series of extracted and normalised verb-subject-object-object frames, as in the following example:

```
Tiger Woods wrapped up the tournament at
four under par.

==(MONTYLINGUA)==>


(Verb: 'wrap up',
 Subj: 'Tiger Woods',
 Obj1: 'tournament',
 Obj2: 'at four under par')
```

When a real-world text document is input into a ConceptNet document-level function, MontyLingua is invoked to extract

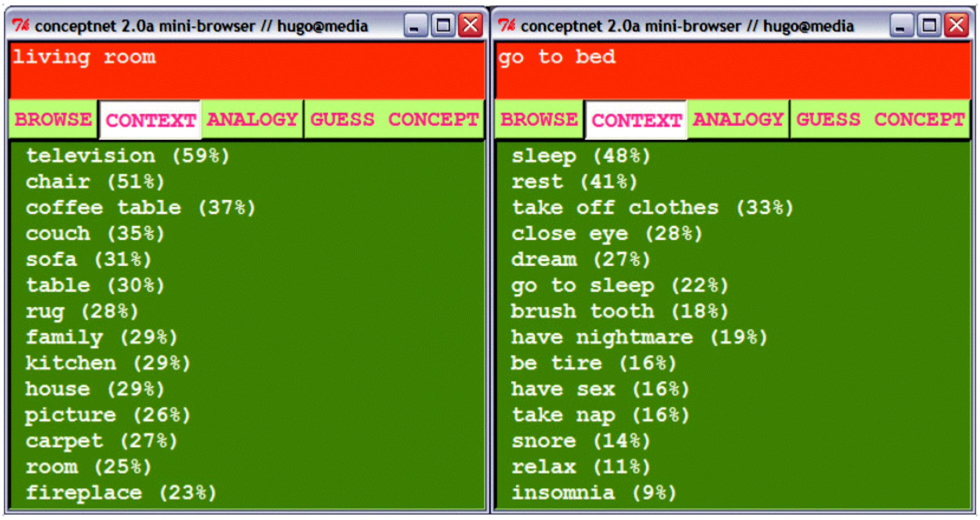

Fig 3 The results of two GetContext() queries are displayed in the ConceptNet knowledge browser.

the verb-subject-object-object frames from the document. These frames closely resemble the syntactically constrained structure of ConceptNet nodes, so reasoning over these frames is a matter of making minor adaptations to fit ConceptNet's needs.

4.2 Contextual neighbourhoods

With all of the complexities associated with the term 'context', we can begin at one very simple notion. Given a concept and no other biases, what other concepts are most relevant? The ConceptNet API provides a basic function for making this computation, called Get Context(). Figure 3 shows ConceptNet's resulting contextual neighbourhood for the concepts 'living room' and 'go to bed'.

A neat property of these results is that they are easy to verify with one's own intuition. While people are known to be very good at this sort of context task, computers are not because they lack the careful, connectionist wiring-together-of-ideas which exists in a person's mind. As a semantic network whose concepts are connected via many dimensions, ConceptNet can begin to approximate simple human capabilities from context.

Technically speaking, the contextual neighbourhood around a node is found by performing spreading activation radiating outward from that source node. The relatedness of any particular node is not simply a function of its link distance from the source, but also considers the number and strengths of all paths which connect the two nodes.

4.2.1 Realm-filtering

Recognising that the relevance of each relation-type varies with respect to each task or application domain, relation-types are assigned a different set of numeric weights for each task. In so doing, spreading activation is nuanced. In the ARIA Photo Agent, Liu et al [10] heuristically weighted each semantic relation type based on their perceived importance to the photo retrieval domain, and then further trained the numerical weights of each relation-type on a domain-specific corpus. In spreading activation, it may also be desirable to turn off certain relation-types altogether. In this manner, we can get temporal, spatial, or action-only neighbourhoods of concepts. We call this realm-filtering. For example, getting only the temporally forward conceptual expansions would be equivalent to imagining possible next states from the current state.

4.2.2 Topic generation

The GetContext() function is useful for semantic query expansion and topic generation. A few novel AI intelligent systems have been built around this simple idea. For example, Musa et al's GloBuddy system [16] is a dynamic foreign-language phrase book that uses ConceptNet's GetContext() feature to generate a collection of phrases paired with their translations on a given topic. For example, entering 'restaurant' would return phrases like 'order food' and 'waiter' and 'menu', and their translations in the target language.

Another way to use GetContext() is for querying the contextual intersection of multiple concepts. If we extract all the concepts from a text document and take their intersection, we can achieve the inverse of topic generation, which is topic gisting. This is discussed in a following subsection.

4.3 Analogy-making

Like context manipulation, analogy-making is another fundamental cognitive task. For people, making analogies is critical to learning and creativity. It is a process of decomposing an idea into its constituent aspects and parts, and then seeking out the idea or situation in the target domain that shares a salient subset of those aspects and parts.

analogy-making is another fundamental cognitive task

Because AI is often in the business of dissecting ideas into representations like schemas and frames [2], analogy-making is quite prevalently used. It goes by pseudonyms like fuzzy matching, case-based reasoning [17], structure-mapping theory [18], and high-level perception [19]. While in principle, a basic form of analogy is easy to compute, AI programs have long lacked the large-scale, domain-general repository of concepts and their structural features required to support commonsensical analogy-making. We believe that ConceptNet serves this need to some approximation.

Gentner's structure-mapping theory of analogy emphasises formal, shared syntactic relations between concepts. In contrast, Hofstadter and Mitchell's 'slipnets' [20] project emphasises semantic similarities and employs connectionist notions of conceptual distance and activation to make analogy more dynamic and cognitively plausible. Analogy in ConceptNet can be coaxed to resemble either structure-mapping or slipnets depending on whether weakly semantic relations (e.g. 'LocationOf', 'IsA') or strongly semantic relations (e.g. 'PropertyOf', 'MotivationOf') are emphasised in the analogy. Analogy in ConceptNet also has a slipnet-like connectionist property in that connections between nodes are heuristically weighted by the strength or certainty of a particular assertion.

Stated concisely, two ConceptNet nodes are analogous if their sets of back-edges (incoming edges) overlap. For example, since 'apple' and 'cherry' share the back-edges, [(PropertyOf x 'red'); (PropertyOf x 'sweet'); (IsA x 'fruit')], they are in a sense, analogous concepts. Of course, it may not be aesthetically satisfying to consider such closely related things analogous (perhaps their shared membership in the set, fruit, disqualifies them aesthetically), but for the purpose of keeping our discussion simple, we will not indulge such considerations here. In Fig 4, we give a screenshot of resulting analogous concepts of 'war', as computed in ConceptNet.

As with the GetContext() feature, it may also be useful to apply realm-filtering to dimensionally bias the GetAnalogousConcepts() feature. We may, for example, prefer to variously emphasise functional similarity versus affective similarity versus attribute similarity by weighting certain relation-types more heavily than others.

4.4 Projection

A third fundamental inference mechanism is projection, which is graph traversal from an origin node, following a single transitive relation-type. 'Los Angeles' is located in 'California', which is located in 'United States', which is located on 'Earth' is an example of a spatial projection, since LocationOf is a transitive relation. A transitive relation is one that is amenable to *modus ponens* reasoning (i.e. IF A→B AND B→C, THEN A→C). In ConceptNet, both containment relation-types (i.e. LocationOf, IsA, PartOf, MadeOf, FirstSubeventOf, LastSubeventOf, SubeventOf), and ordering relation-types (i.e. EffectOf, DesirousEffectOf) are transitive, and can be leveraged for projection.

Subevent projection may be useful for goal planning, while causal projection may be useful for predicting possible outcomes and next-states. Liu and Singh's MAKEBELIEVE system [21], for example, is an interactive storytelling system that can generate simple English stories, using OMCS causal projection to ponder different plot-lines. Wang's SAM Collaborative Storytelling Agent [22] also used causal projection in ConceptNet's predecessor system to drive the selection of discourse transitions.

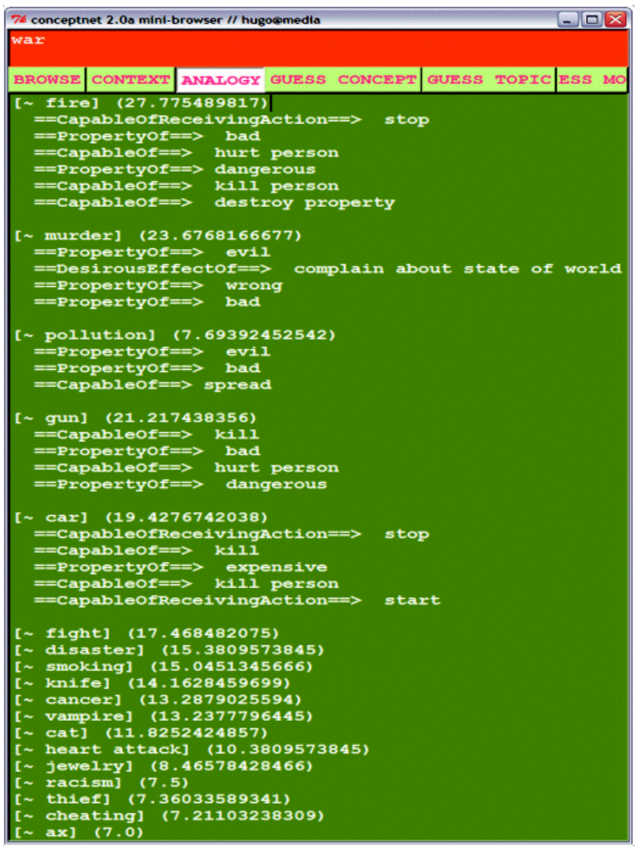

Fig 4 The results of a GetAnalogousConcepts() query for 'war' are displayed in the ConceptNet knowledge browser. Structures shared in the analogy are only shown for the first five concepts.

4.5 Topic gisting

Topic gisting is a straightforward extension of the GetContext() feature to accept the input of real-world documents. Its value to information retrieval and data mining is immediately evident.

Using MontyLingua, a document is gisted into a sequence of verb-subject-object-object (VSOO) frames. Minor transformations are applied to each VSOO frame to massage concepts into a ConceptNet-compatible format. These concepts are heuristically assigned saliency weights based on lightweight syntactic cues, and their weighted contextual-intersection is computed by GetContext().

GetContext() used in this way serves as a naïve topic spotter. To improve performance it may be desirable to designate a subset of nodes to be more suitable as topics than others. For example, we might designate 'wedding' as a better topic than 'buy food' since ConceptNet has more knowledge about its subevents (e.g. 'walk down aisle', 'kiss bride'), and its parts (e.g. 'bride', 'cake', 'reception').

Previous to the addition of this feature to ConceptNet, Eagle et al [12] used GetContext() in a similar fashion to gist topics from overheard conversations. Researchers in text summarisation such as Hovy and Lin have recognised the need for symbolic general world knowledge in topic detection, which is a key component of summarisation. In SUMMARIST [23], Hovy and Lin give the example that the presence of the words 'gun', 'mask', 'money', 'caught', and 'stole' together would indicate the topic of 'robbery'. However, they reported that WordNet and dictionary resources were relationally too sparse for robust topic detection. ConceptNet excels at this type of natural language contextual task because it is relationally richer and contains practical rather than dictionary-like knowledge.

Inspired by Hovy and Lin's example, Fig 5 depicts a visualisation of the output of ConceptNet's topic-gisting function as applied to the four input concepts of 'accomplice', 'habit', 'suspect' and 'gun'.

4.6 Disambiguation and classification

A task central to information management is the classification of documents into genres (e.g. news, spam), and a task central to natural-language-processing is the disambiguation of the meaning of a word given the context in which it appears (e.g. in 'Fred ate some chips', are the chips 'computer chips' or 'potato chips?'). A naïve solution to classification and disambiguation is implemented in ConceptNet. For each class or disambiguation-target, an exemplar document is fed into a function that computes the contextual-regions they occupy in the ConceptNet semantic network. New documents are classified or disambiguated into the exemplars by calculating the nearest neighbour.

This approach is similar to the ones taken by statistical classifiers which compute classification using cosine-distance in high-dimensional vector space. The main difference in our approach is that the dimensions of our vector space are commonsense-semantic (e.g. along dimensions of time, space, affect) rather than statistically based (e.g. features such as punctuation, keyword frequency, syntactic role).

4.7 Novel-concept identification

A critical application of analogy-making is learning the meanings of novel or unknown concepts. To explain what a 'potsticker' or 'dumpling' is to someone who has never had one, it might be a good strategy to draw comparison to more familiar concepts like 'ravioli' (i.e. calling ravioli's structure to mind) or describe its composition (e.g. PartOf, MadeOf), or perhaps that you can eat it (e.g. UsedFor, CapableOf ReceivingAction), order it in a Chinese restaurant (e.g. LocationOf), or that it is hot and delicious (e.g. PropertyOf). Novel-concept identification can also be useful to information systems. It might, for example, allow a person to search for something whose name cannot be recalled, or facilitate the disambiguation of pronouns based on their semantic roles. In the ConceptNet API, GuessConcept() takes as input a document and a novel concept in that document. It outputs a list of potential things where the novel concept might be by making analogies to known concepts.

4.8 Affect sensing

ConceptNet's API function, GuessMood(), performs textual affect sensing over a document. The algorithm is a simplification of Liu et al's Emotus Ponens system [24].

Its technical workings are quite easily described. Consider that a small subset of the concepts in ConceptNet are first affectively classified into one of six affect categories (happy, sad, angry, fearful, disgusted, surprised).

The affect of any unclassified concept can be assessed by finding all the paths which lead to each of these six affectively known categories, and then judging the strength and frequency of each set of paths. GuessMood() is a more specialised version of ConceptNet's Classification function.

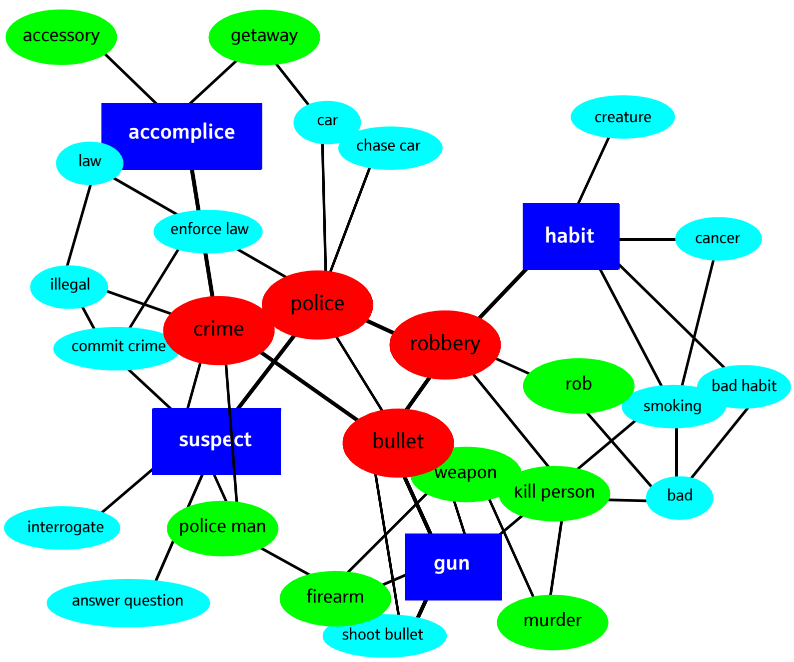

Fig 5 Computer-generated visualisation shows a portion of results from a ConceptNet topic-gisting query. Rectangular nodes represent the concepts from the input document. Red ovals are most relevant output topics, with relevance decreasing from green ovals to light blue ovals.

In summary, we have described how the ConceptNet tool-kit supports various contextual commonsense-reasoning tasks. At present, three node-level functionalities are implemented, context-finding, analogy-making, and projection, as well as four document-level functions, topic-gisting, disambiguation and classification, novel-concept identification, and affect sensing. Each of these contextual reasoning functions benefits common information management and natural-language-processing tasks; furthermore, they go beyond the needs of many existing applications to suggest new AI-based intelligent systems.

Of course, the utility of ConceptNet's reasoning abilities hinge largely on the quality of the knowledge it contains. In the following section, we ponder the question: 'Are the contents of ConceptNet any good?'

5. Characteristics and quality of the ConceptNet knowledge base

Large knowledge bases of commonsense knowledge like ConceptNet are somewhat difficult to evaluate. What is and is not 'common sense?' What are optimal ways to represent and reason with 'common sense?' How does one assess the goodness of knowledge that is defeasible and expressible in varying ways? How much commonsense about a topic or concept constitutes completeness? These are all difficult questions that we cannot provide definitive answers for. One important criterion driving the evolution of ConceptNet has been: 'Is it usable and how is it improving the behaviour of the intelligent system in which it is being applied?' Section 6 makes an attempt to answer this question by reviewing applications built on ConceptNet, many of which have themselves been evaluated.

ConceptNet's reasoning abilities hinge largely on the quality of its knowledge

In this section, we attempt to characterise very broadly the coverage and goodness of the knowledge base as a whole. We approach the issue of coverage by making some quantitative inquiries into the ConceptNet knowledge base. Our discussion of goodness looks at some human evaluations of OMCS and ConcepNet.

5.1 Characteristics of the knowledge base

Figure 2 illustrated the distribution of the knowledge base according to relation-type. This informs us about ConceptNet's areas of expertise and weakness. Roughly half of what ConceptNet knows (excluding k-lines) concerns abilities and functions.

We might also want to know about the complexity of ConceptNet's nodes. Are concepts expressed simply or obscurely? A simple (but telling) statistic is the histogram of nodal word-lengths. The shorter the nodes, the less complex they are likely to be. These results are given in Fig 6. Approximately 70% of the nodes have a word-length of less than or equal to three. Since a verb-noun_phrase-

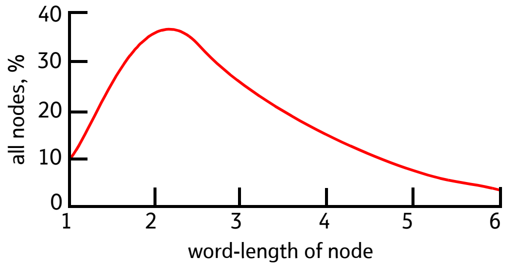

Fig 6 Examining the histogram of nodal word-lengths gives us a clue as to the likely complexity of nodes in ConceptNet.

prepositional_phase compound (e.g. 'take dog for walk') requires at least four words, we know that the complexity of the vast majority of nodes is syntactically less complex that this. Also, the 50% of nodes with a word-length of one or two are likely to be atomic types (e.g. noun phrase, prepositional phrase, adjectival phrase) or the simplest verb-noun compounds (e.g. 'buy book'). These are all relatively non-complex types. If ConceptNet's concepts are generally not very structurally complex, does that mean that most assertions are simple, and thus, have repeated utterances? To answer this question, we calculate the frequency with which ConceptNet's unique assertions are uttered in the OMCS corpus (Fig 7), and the frequency with which one assertion can be inferred from other assertions. Inferred assertions, an indirectly stated kind of knowledge, can be thought of as 'echoes' of uttered assertions.

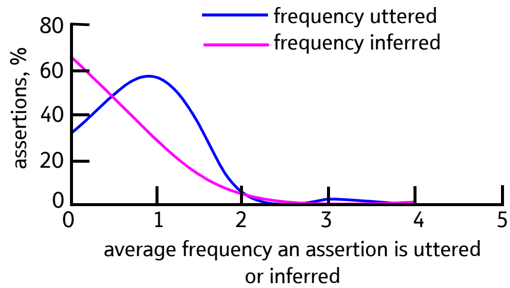

Fig 7 Assessing the strength of ConceptNet assertions by examining how many times each assertion is uttered and/or inferred.

Figure 7 reveals that roughly 32% of assertions are never uttered (purely inferred, these are all k-lines) and 58% of assertions are uttered only once, leaving 10% (160 000 assertions) which are uttered two or more times. If we disregard the unuttered k-line knowledge, then 85% of assertions are uttered once and 15% more than once. While most assertions (65%) have no 'echoes' (inferred elsewhere), 25% have one echo, and 10% have two or more echoes. Not shown in Figure 7 is that 18% of the assertions (300 000 assertions) have an uttered-inferred combined frequency of two or greater, which can be taken as a positive indication of commonality.

Despite the fact that 70% of nodes have three or fewer words, still 90% of assertions are uttered zero times or only one time. It is somewhat surprising that there is not more overlap, but this speaks dually to the broadness of the space of 'commonsense', and to the great variation introduced by our natural language node representation. Still, we defend the fact that natural language allows the same idea to be expressed slightly differently in many ways. These variations are not wasted effort. Each choice of verb, adjective, and noun phrase creates a psychological context which provides nuances on the concept's interpreted meaning. The maintenance of surface variations also assists in mapping nodes on to real-world documents.

To improve the commonality and convergence of the knowledge, we should focus on improving the relaxation phase in which lexical resources help to reconcile nodes. We have only scratched the surface here. It is somewhat encouraging that while only 10% of assertions are uttered more than once, 18% of assertions have a combined utterance-echo count of more than one. Relaxation assists in convergence by finding echoes that corroborate and strengthen uttered assertions, and there is much potential for improvement in this regard.

A final characterisation of the knowledge base examines the connectivity of the semantic network by measuring nodal edge-density (Fig 8). This data speaks quite positively of the dataset. With the addition of k-line knowledge, nodal edge-densities increase quite favourably, with 65% of nodes having two or more links, and 45% of nodes having three or more links. This either means that k-lines are very well-connected among themselves, or that k-lines mainly facilitate the connectivity of nodes otherwise already connected. The truth is probably a mix of the two extremes. In any case, the importance of a well-connected network to machinery that purports to reason about context cannot be understated.

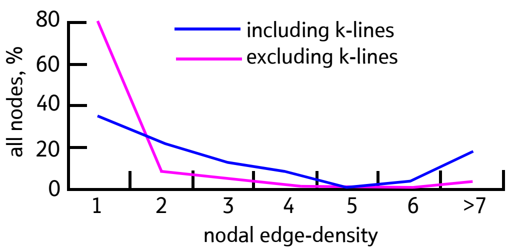

Fig 8 The connectivity of nodes in ConceptNet is illustrated by a histogram of nodal edge-densities. The addition of k-lines effects a marked improvement on network connectivity.

5.2 *Quality of the knowledge*

Since ConceptNet derives from the Open Mind Common Sense corpus, it is relevant to talk about the quality of that body of knowledge. The original OMCS corpus was previously evaluated by Singh et al [6]. Human judges evaluated a sample of the corpus and rated 75% of items as largely true, 82% as largely objective, 85% as largely making sense, and 84% as knowledge someone would have by high school.

We have also evaluated the knowledge in ConceptNet; however, the evaluation was performed not over the current dataset, but over a dataset circa 2003. As a result, k-line knowledge is absent and remains unevaluated. The basic extraction algorithms have not changed significantly, and if anything, we suggest that the quality (and computability) of knowledge has improved in version 2.0 over previous versions such as version 1.2, which was the subject of the evaluation. Since version 1.2, we have implemented better noise filtering on nodes by employing syntactic and semantic constraints. The evaluation of version 1.2 is given below for completeness.

5.2.1 Evaluation of ConceptNet version 1.2

We conducted an experiment with five human judges and asked each judge to rate 100 concepts in ConceptNet version 1.2 — 10 concepts were common to all judges (for correlational analysis), 90 were of their choice. If a concept produced no results, they were asked to duly note that and try another concept. Concepts were judged along these two dimensions, each on a Likert 1 (strongly disagree) to 5 (strongly agree) scale:

- results for this concept are fairly comprehensive,
- results for this concept include incorrect knowledge, nonsensical data, or non-commonsense information.

To account for inter-judge agreement, we normalised scores using the ten common concepts, and produced the re-centred aggregate results shown below in Table 2.

Table 2 Two dimensions of quality of ConceptNet, rated by human judges.

	Mean score	Standard Deviation	Standard Error
Comprehensiveness	3.40/5.00	1.24	1.58
Noisiness	1.24/5.00	0.99	1.05%
Concepts attempted, but not in ConceptNet	11.3%	6.07%	0.37%

These results can be interpreted as follows. With regard to comprehensiveness, ConceptNet's concepts were judged as containing, on average, several relevant concepts, but varied significantly from a few concepts to almost all of the concepts. ConceptNet's assertions were judged to have little noise on average, and did not vary much. Roughly one out of every ten concepts chosen by the judges were missing from ConceptNet. We are optimistic about these results. Comprehensiveness was moderate but varied a lot, indicating that coverage of commonsense topic areas is still patchy, which we hope will improve as OMCS grows (though perhaps acquisition should be directed into poorly covered topic areas). Noisiness was surprisingly low, lending support to the idea that a relatively clean knowledge base can be elicited from public acquisition. The percentage of knowledge base misses was more than tolerable considering that ConceptNet version 1.2 had only 45 000 natural language concepts — a tiny fraction of those possessed by people.

It is not clear how indicative this type of human evaluation is. Evaluations such as these are fundamentally problematic in that, when asked to choose 'commonsense' concepts, a stereotype is invoked, possibly preventing a judge from remembering anything but the most glaring examples which fit the prototype of what 'commonsense' is. This sort of self-reporting bias returns us to the problem of finding suitable ways to evaluate ConceptNet's coverage and goodness.

While it is difficult to attain a global assessment of ConceptNet's coverage and quality, it is easier to measure coverage and goodness against a system's performance in concrete tasks and applications. In the following section, we culminate our discussion on evaluation by suggesting that the gamut of applications that have been built using the ConceptNet tool-kit, many of which have themselves been evaluated, be considered as a corpus of application-specific evaluation.

6. Applications of ConceptNet

If the purpose of evaluating a resource is meant to help us decide whether or not the resource can be applied to solve a problem, then certainly there is evaluative merit in the fact that ConceptNet has been driving tens of interesting research applications since 2002. Many of these research applications were completed as final term projects for a commonsense reasoning course that was taught at the MIT Media Lab. Some of ConceptNet's more interesting applications are enumerated below. For a more judicious treatment of ConceptNet's applications please refer to Lieberman et al [11, 25].

6.1 Commonsense ARIA

Commonsense ARIA [9] observes a user writing an e-mail and proactively suggests photos relevant to the user's story. The photo annotation expansion system, CRIS (ConceptNet's oldest predecessor) bridges semantic gaps between annotations and the user's story (e.g. 'bride' and 'wedding').

6.2 GOOSE

GOOSE [26] is a goal-oriented search engine for novice users. Taking in a high-level goal description, e.g. 'I want to get rid of the mice in my kitchen', GOOSE combines commonsense inference and search expertise to generate the search query, 'pest control' 'cambridge, ma'.

6.3 MAKEBELIEVE

MAKEBELIEVE [21] is story-generator that allows a person to interactively invent a story with the system. MAKEBELIEVE uses a ConceptNet predecessor to generate causal projection chains to create storylines.

6.4 GloBuddy

GloBuddy [15] and GloBuddy 2 [11] is a dynamic foreign language phrasebook which, when given a situation like 'I am at a restaurant', automatically generates a list of concepts relevant to the situation like 'people', 'waiter', 'chair', and 'eat' and their corresponding translations.

6.5 AAA — a profiling and recommendation system

AAA [13] recommends products from Amazon.com by using ConceptNet to reason about a person's goals and desires, creating a profile of their predicted tastes.

6.6 OMAdventure

OMAdventure [13] is an interactive scavenger hunt game where players navigate a dynamically generated graphical world.

6.7 Emotus Ponens

Emotus Ponens [24] is a textual affect-sensing system that leverages commonsense to classify text using six basic emotion categories. EmpathyBuddy is an e-mail client which gives the author automatic affective feedback via an emoticon face.

6.8 Overhear

Overhear [12] is a speech-based conversation understanding system that uses commonsense to gist the topics of casual conversations.

6.9 Bubble Lexicon

Bubble Lexicon [27] is a context-centred cognitive lexicon that gives a dynamic account of meaning. ConceptNet bootstraps the lexicon's connectionist-semantic network with world semantic knowledge.

6.10 LifeNet

LifeNet [28] is a probabilistic graphical model of everyday first-person human experience. LifeNet is built by reformulating ConceptNet into egocentric propositions (e.g. (EffectOf 'drink coffee', 'feel awake')… ('I drink coffee'… 'I feel awake'), and linking them together with transition probabilities.

6.11 SAM

SAM [22] is an embodied storytelling agent that collaboratively tells stories with children as they play with a doll's house. ConceptNet drives SAM's choice of discourse transitions.

6.12 What Would They Think?

'What Would They Think?' [29] automatically models a person's personality and attitudes by analysing personal texts such as e-mails, weblogs, and homepages. ConceptNet's analogy-making is used to make attitude-prediction more robust.

6.13 Commonsense Predictive Text Entry

'Commonsense Predictive Text Entry' [30] leverages ConceptNet to understand the context of a user's mobile-phone text-message and to suggest likely word completions.

6.14 Commonsense Investing

Commonsense Investing [31] assists personal investors with financial decisions by mapping ConceptNet's representation of a person's goals and desires into an expert's technical terms.

6.15 Metafor

Metafor [32, 33] facilitates children in exploring programming ideas by allowing them to describe programs using English. ConceptNet provides a programmatic library of 'commonsense classes' used for the programmatic-semantic interpretation of natural language input

7. Conclusions

ConceptNet is presently the largest freely available database of commonsense knowledge. It comes with a knowledge browser and an integrated natural-language-processing engine that supports many practical textual-reasoning tasks including topic generation, topic gisting, semantic disambiguation and classification, affect sensing, analogy making, and other context-oriented inferences.

ConceptNet is designed to be especially easy to use; it has the simple structure of WordNet and its underlying representation is based on natural language fragments, making it particularly well suited to textual-reasoning problems. Motivated by the range of concepts available in the Cyc commonsense knowledge base, the content of ConceptNet reflects a far richer set of concepts and semantic relations than those available in WordNet. While the coverage of ConceptNet's knowledge is still spotty in comparison to what people know, our analysis has shown it to be surprisingly clean, and it has proved more than large enough to enable experimenting with entirely new ways of tackling traditional semantic processing tasks.

Whereas WordNet excels at lexical reasoning, and Cyc excels at precise logical reasoning, ConceptNet's forte is contextual commonsense reasoning — a research area that is poised to redefine the possibilities for intelligent information management. Since 2002, ConceptNet has powered tens of exciting and novel research applications, many of which were engineered by undergraduates in a school semester. We think that this speaks volumes to ConceptNet's uniquely simple engineering philosophy — giving a computer common sense need not require volumes of specialised knowledge in AI reasoning and natural language processing. We envision this project as being a part of a new commonsense AI research agenda — one that is grounded in developing novel real-world applications which provide great value, and whose implementation would not be possible without resources such as ConceptNet. We hope that this paper has encouraged the reader to consider using ConceptNet within their own projects, and to discover the benefits afforded by such large-scale semantic resources.

Acknowledgments

We would like to thank the many people at the Media Lab who have used ConceptNet in their projects, especially Barbara Barry, Henry Lieberman, and Ian Eslick. We would also like to thank the students in Henry Lieberman's Common Sense Reasoning for Interactive Applications course for making the most of early versions of the ConceptNet tool-kit, and the over 14 000 users of the Open Mind Common Sense Web site for contributing their time and effort to our project. Finally, we thank the blind reviewers for their thoughtful feedback.

References

1 Liu H and Singh P: 'Commonsense reasoning in and over natural language', Proceedings of the 8th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES-2004) (2004).

2 Minsky M: 'The society of mind', Simon & Schuster (1987).

3 Dreifus C: 'Got stuck for a moment: an interview with Marvin Minsky', International Herald Tribune (August 1998).

4 Fellbaum C (Ed): 'WordNet: An Electronic Lexical Database', MIT Press (1998).

5 Lenat D B: 'CYC: A large-scale investment in knowledge infrastructure', Communications of the ACM, 38, No 11, pp 33—38 (1995).

6 Singh P, Lin T, Mueller E T, Lim G, Perkins T and Zhu W L: 'Open mind commonsense: 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 No 2519 Heidelberg, Springer (2002).

7 Turner E: 'OMCSNet-WNLG Project', (2003) — http://www.eturner.net/omcsnetcpp/wordnet

8 Gelernter D: 'The Muse in the Machine: Computerising the Poetry of Human Thought', Free Press (1994).

9 Collins A and Loftus E: 'A spreading-activation theory of semantic processing', Psychological Review, 82, No 6, pp 407—428 (1975).

10 Liu H and Lieberman H: 'Robust photo retrieval using world semantics', Proceedings of LREC2002 Workshop: Using Semantics for IR, Canary Islands, pp 15—20 (2002).

11 Lieberman H, Faaborg A, Espinosa J and Stocky T: 'Commonsense on the go', BT Technol J, 22, No 4, pp 241—252 (October 2004).

12 Eagle N, Singh P and Pentland A: 'Common sense conversations: understanding casual conversation using a common sense database', Proceedings of the Artificial Intelligence, Information Access, and Mobile Computing Workshop (IJCAI 2003) (2003).

13 Commonsense reasoning for interactive applications projects (2003) — http://www.media.mit.edu/~lieber/Teaching/Common-Sense-Course/Projects/Projects-Intro.html

14 Liu H: 'MontyLingua v1.3.1', Toolkit and API (2003) — http://web.media.mit.edu/~hugo/montylingua/

15 Baker C F, Fillmore C J and Lowe J B: 'The Berkeley FrameNet project', in Proceedings of the COLING-ACL, Montreal, Canada (1998).

16 Musa R, Scheidegger M, Kulas A and Anguilet Y: 'GloBuddy, a dynamic broad context phrase book', in Proceedings of CONTEXT'2003, pp 467—474, Springer (2003).

17 Leake D B: 'Case-based reasoning: experiences, lessons, and future directions', Menlo Park, California, AAAI Press (1996).

18 Gentner D: 'Structure-mapping: A theoretical framework for analogy', Cognitive Science, 7, pp 155—170 (1983).

19 Chalmers D J, French R M and Hofstadter D R: 'High-level perception, representation, and analogy: a critique of artificial intelligence methodology', Technical Report CRCC-TR-49, Center for Research in Concepts and Cognition, Indiana University (March 1991).

20 Hofstadter D R and Mitchell M: 'The copycat project: A model of mental fluidity and analogy-making', in Hofstadter D and the Fluid Analogies Research group: 'Fluid Concepts and Creative Analogies', Basic Books, Chapter 5, pp 205—267 (1995).

21 Liu H and Singh P: 'MAKEBELIEVE: using commonsense to generate stories', Proceedings of the Eighteenth National Conference on Artificial Intelligence, AAAI Press, pp 957—958 (2002).

22 Wang A: 'Turn-taking in a collaborative storytelling agent', Masters Thesis, MIT Department of Electrical Engineering and Computer Science (2002).

23 Hovy E H and Lin C H: 'Automated text summarisation in SUMMARIST', Proceedings of the ACL97/EACL97 Workshop on Intelligent Scalable Text Summarisation, Madrid, Spain (July 1997).

24 Liu H, Lieberman H and Selker T: 'A model of textual affect sensing using real-world knowledge', in Proceedings of IUI 2003, Miami, Florida (2003).

25 Lieberman H, Liu H, Singh P and Barry B: 'Beating some common sense into interactive applications', AI Magazine (to appear in Fall 2004 issue).

26 Liu H, Lieberman H and Selker T: 'GOOSE: a goal-oriented search engine with commonsense', in Bra D and Conejo B (Eds): 'Adaptive Hypermedia and Adaptive Web-based Systems', Second

International Conference (AH'2002), Proceedings. Lecture Notes in Computer Science No 2347, Springer, pp 253—263 (2002).

27 Liu H: 'Unpacking meaning from words: A context-centered approach to computational lexicon design', in Blackburn et al (Eds): 'Modeling and Using Context', 4th International and Interdisciplinary Conference (CONTEXT 2003) Proceedings, Lecture Notes in Computer Science No 2680, Springer, pp 218—232 (2003).

28 Singh P and Williams W: 'LifeNet: a propositional model of ordinary human activity', Proceedings of the Workshop on Distributed and Collaborative Knowledge Capture (DC-KCAP) at K-CAP 2003, Sanibel Island, Florida (2003).

29 Liu H and Maes P: 'What Would They Think? A computational model of attitudes', Proceedings of the ACM International Conference on Intelligent User Interfaces (IUI 2004), Madeira, Funchal, Portugal, ACM2004, pp 38—45 (January 2004).

30 Stocky T, Faaborg A and Lieberman H: 'A Commonsense approach to predictive text entry', Proceedings of Conference on Human Factors in Computing Systems (CHI'04), Vienna, Austria (2004).

31 Kumar A, Sundararajan S and Lieberman H: 'Commonsense investing: bridging the gap between expert and novice', Proceedings of Conference on Human Factors in Computing Systems (CHI 04), Vienna, Austria (2004).

32 Liu H and Lieberman H: 'Toward a programmatic semantics of natural language', Proceedings of VL/HCC'04: the 20th IEEE Symposium on Visual Languages and Human-Centric Computing, pp 26—29, Rome, IEEE Computer Society Press (2004).

33 Lieberman H and Liu H: 'Feasibility studies for programming in natural language', in Lieberman H, Paterno F and Wulf V (Eds): 'Perspectives in End-User Development', Kluwer (Summer 2004).

Hugo Liu is a third-year PhD candidate in the Interactive Experience group at the Media Lab. He researches humanesque cognition from an artificial intelligence standpoint. His research includes cognition-inspired computational modelling of everyday reasoning, emotions, personality and attitudes, hermeneutics and story understanding, and information foraging. He has contributed two freely available and widely used toolkits to the AI research community-at-large: ConceptNet, a practical, commonsense reasoning tool-kit, and MontyLingua, an end-to-end natural language understander for English.

Push Singh is a doctoral candidate in MIT's Department of Electrical Engineering and Computer Science, and conducts his doctoral research as a member of Marvin Minsky's Society of Mind group at the Media Lab, where his work is focused on finding ways to give computers human-like commonsense. He created the Open Mind Common Sense Web site to acquire a corpus of commonsense statements from the general public; a semantic network called OMCSNet was mined from this corpus, and has been used for a number of research projects at MIT. He received his BS and MEng in electrical engineering and computer science from MIT.