An Interface for Crowd-Sourcing Spatial Models of Commonsense

Benjamin Johnston
Innovation and Enterprise Research Laboratory
Faculty of Engineering and Information Technology
University of Technology, Sydney
Australia 2007

Abstract

Commonsense is a challenge not only for representation and reasoning but also for large scale knowledge engineering required to capture the breadth of our 'everyday' world. One approach to knowledge engineering is to 'outsource' the effort to the public through games that generate structured commonsense knowledge from user play. To date, such games have focused on symbolic and textual knowledge. However, an effective commonsense reasoning system will require spatial and physical reasoning capabilities. In this paper, I propose a tool for gathering commonsense information from ordinary people. It is a user-friendly 3D sculpting tool for modeling and annotating models of physical objects and spaces.

Introduction

There are many applications of commonsense reasoning that involve spatial information:

- Word sense disambiguation. For example, the ambiguity in the sentence 'the pig is in the *pen*' might be resolved by comparing the size of a pig to the volume inside a ball-point pen and a pigpen.
- Planning. For example, the problem of planning how to transport a sofa depends on knowledge of the size of a typical door-frame, the capacity of a car and ways of securing the object to a truck.
- Vision and interpretation. For example, a domestic robot that is asked, 'fetch me the coffee on the table,' will need to know the shape of a table and coffee cup to be able to recognize them, will need to know what part of a table to look for objects that are said to be 'on' the table and may need to understand the spatial associations with coffee and kitchen tables versus a table in the bathroom.

Unfortunately, such knowledge does not lend itself to elegant and self-contained axiomatization. A logical or symbolic system that is capable of spatial commonsense reasoning requires a huge volume of specific facts to be explicitly coded into its knowledge base. These facts might be encoded as axioms that, for example, state the typical dimensions of a pig, dictate that a table ordinarily has four legs attached to the underside of a flat surface and that an ordinary sedan is not able to carry a typical sofa inside.

Even though logical formalizations are well suited to capturing the abstract properties of space and naïve physics, the breadth of commonsense is such that the axiomatic methods of modeling knowledge in logical formalisms need to be augmented with techniques for large-scale data collection. The potential number of facts or axioms is limitless; there are countless simple questions about the real world that cannot be answered by pure inference from a small kernel of knowledge. They must be learnt or taught. The only guard against unanticipated queries is a complete formal description of every object in the universe of discourse. However, the Cyc project shows that direct formalization can be very difficult and expensive (Wood 2005).

The approach used in creating ConceptNet (Havasi, Speer and Alonso 2007) is one solution to covering the breadth of commonsense. ConceptNet is a network of informal semantic knowledge, structured around 21 relations (*ibid.*). In ConceptNet, formal precision is sacrificed to gain broader coverage. By sacrificing precision, knowledge can be collected from the efforts of volunteers (rather than skilled engineers) playing specially crafted games on the internet. The instances in ConceptNet's relations (such as IsA, MadeOf, UsedFor, CapableOf, DesireOf and InstanceOf) correspond to imprecise human intuitions rather than a more objective truth. For example, in ConceptNet, both tomatoes and mushrooms are vegetables, even though some would argue that they are fruits and fungi respectively, rather than

vegetables. Nevertheless, such informal knowledge has been demonstrated in useful applications (*e.g.*, (Shen, Lieberman and Lam 2007; Faaborg *et al.* 2005)). It can furthermore be combined with formal logics of commonsense to, for example, deduce that an agent is likely to perform an action (Performs) given that it is capable of the action (CapableOf) and desires the outcome (DesireOf).

While ConceptNet is a useful knowledgebase of simple relationships between concepts, it lacks spatial knowledge apart from the three relations PartOf, HasA, AtLocation and LocatedNear. The objective of this paper is, therefore, to develop a representation scheme and tools suitable for collecting broad commonsense knowledge of the physical and spatial world.

In this paper, I propose a simple homogenous representation suitable for capturing a 'first-order' approximation of rich (but informal) physical knowledge. While the representation is relatively informal and lacks the expressiveness of first order logic, it serves as a practical compromise between having a narrow but highly expressive logic and not having any physical knowledge at all. In fact, it is intended that the scheme be used alongside expressive logics; complementing the depth of a formal ontology with the breadth of a large knowledgebase of instances.

I then proceed to describing preliminary work on a user-friendly tool that enables ordinary people to participate in the creation of commonsense knowledge. The objective of this tool is to, like ConceptNet, 'crowd-source' the knowledge engineering process across a huge number of small contributors.

Pragmatic Representations of Objects

I previously proposed Comirit as a hybrid reasoning framework for combining logical knowledge with simulations (Johnston and Williams 2007). In Comirit, physical objects are modeled using a large mass-spring system. The shape and physical properties of objects are approximated by a set of point-masses that are connected to each other by springs. Thus, physical reasoning problems involving 'forward inference' from a set of initial conditions can be performed by iteratively solving the laws of physics over the simulation.

This technique may be viewed as a polynomial-time approximation to an axiomatization of the laws of physics. In those reasoning problems where approximate forward inference is *sufficient*, such simulations may entirely obviate the need for logical reasoning from principles of physics or naïve physics. Furthermore, the representation is well suited to creating large knowledge bases: the consistent representation allows for standardized tools that manipulate the homogenous representation scheme.

Given this representation scheme, how can large knowledge-bases of physical objects be constructed? ConceptNet contains over 700,000 symbolic assertions for 150,000 concepts (Havasi, Speer and Alonso 2007) created by user submissions. To match this kind of scale in a public repository of commonsense 3D models would be an enormous undertaking. A single 3D model can take anywhere from a few hours to days to build, so a team of 100 might require several years to build just 100,000 models.

ConceptNet was 'crowd-sourced' from volunteers contributing knowledge on a webpage and via online computer games. If 3D modeling could be packaged into an entertaining and intuitive experience, it is not unreasonable to imagine 100,000 volunteers contributing a single 3D model each.

A Simplified Representation Scheme

In building a modeling tool to support hundreds of thousands of users and models, it makes sense to ask, 'what is the simplest 3D representation scheme that could possibly be useful for commonsense reasoning?'

Such a representation scheme should, at least, meet these conditions:

- It should be conceptually clear so that users do not require extensive training (*i.e.*, the steep learning curves associated with existing 3D modeling tools are inappropriate)
- It should be efficiently modeled in popular simulation engines, such as those used by computer games (*i.e.*, this means that it should be closely related to geometric primitives, rather than mass-spring systems)
- It should be efficiently rendered on a users' display
- It should be both easy and efficient to manipulate, transform and analyze
- It should, nevertheless, be capable of describing useful properties of a wide range of objects

Given these objectives, I propose a simple scheme based on connected cubes. That is, we represent an object by a set of voxels. Each voxel is a 3-tuple of integer coordinates that corresponds to a cube aligned to a grid in 3D space. Associated with a set of voxels is a size parameter and a mass parameter. The value of the size parameter is the real world length of the maximum extent of the model along one of the three coordinate axes (*i.e.*, the maximum of the x-length, y-length and z-length). The mass gives the total mass of the object.

In addition to the geometric shape described by voxels, there is an associated set of annotations. Each annotation

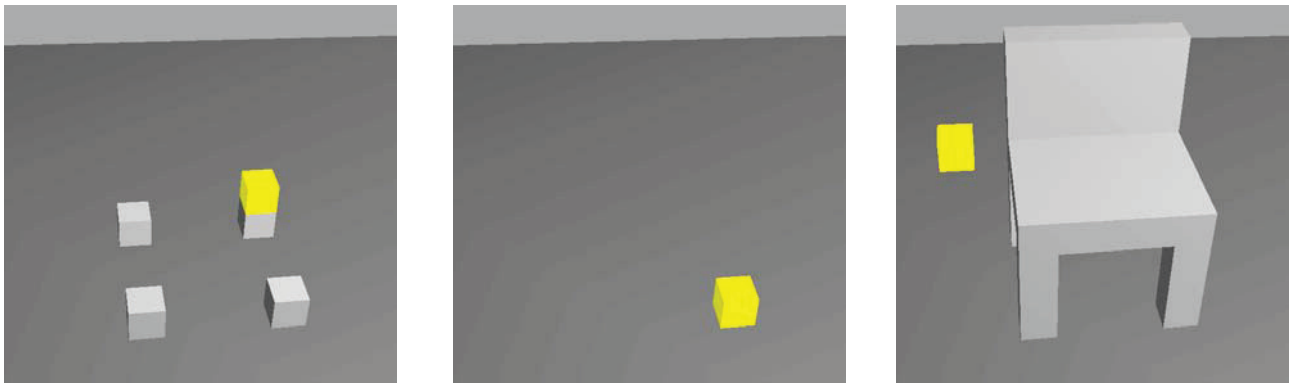

Figure 1. ***(a)*** *The cursor in the 3D modeling tool.* ***(b)*** *The result after creating four voxels.* ***(c)*** *A simple model of a chair.*

relates a voxel to a word or character string. The string is intended to denote a concept or symbol, similar to the informal concepts of ConceptNet. The annotation may be a voxel that is part of the geometric shape or it can be a voxel in empty space. For example, one might annotate the space in the middle of the bucket with the label "contents". While the annotations are the same kind of informal concepts as used in ConceptNet, annotations may also be used to denote formal symbols in a logical language.

More formally, we define a model of an object as the tuple $O = (V, S, M, A)$, where V is a set of integer coordinates $\mathbb{Z}\times\mathbb{Z}\times\mathbb{Z}$ (voxels), S is the real-world length of the longest extent of the object in meters, M is the mass of the object in kilograms and A is a partial relation from the set of integer coordinates to strings $\mathbb{Z}\times\mathbb{Z}\times\mathbb{Z}\leftrightarrow S$. Adjacent voxels are implicitly joined to each other as a fixed solid.

Note that this representation scheme is only used for describing objects. It does not describe the geometric space of a simulation. In fact, it is preferable that space be modeled as a real continuum rather than voxels. That is, even though this representation requires that individual objects within the world are formed from cubes internally aligned in a 3-dimensional grid, the grids of separate objects do not need to align. Separate objects may orient and collide with each other in any direction.

The size parameter allows objects to be modeled at different scales. A high-rise building might be modeled simplistically as, for example, eight cubes stacked on top of each other with a size parameter of 100m. Conversely, a chair might be modeled more intricately from 200 small cubes with the total size parameter being 1.2m. Again, both models may be used simultaneously; they do not require a homogenous grid in simulation but may be used side-by-side with different cube sizes in the same simulation.

A 3D Modeling Tool

An advantage of a simple voxel-based representation is that it lends itself to an extremely simple user interface. I have designed an interface that allows a user to create a 3D model by moving a cursor within a 3D space and setting (or deleting) voxels at the current cursor position. The user interacts with the system using the keyboard and mouse. The arrow keys move the cursor in the *x*-*y* plane, while the page-up and page-down keys move the cursor along the *z* axis. The insert and delete keys are used to set and delete (respectively) voxels at the current cursor position. The mouse can be used to rotate the viewpoint and zoom into parts of the object to check that the shape of the object is correct. Annotations are added at the current cursor location by typing with the keyboard; those annotated voxels are depicted using a secondary color.

A simple prototype has been programmed using the Panda3D game engine (Goslin and Mine 2004). Panda3D is a game engine for Python and C++ that includes a rich API for creating attractive and efficient 3D graphics, in addition to offering a built-in physics simulator (called ODE). Screen-shots of the prototype appear in Figures 1(a)–(c) and Figure 2.

This prototype is designed to be very 'discoverable'. During informal usability tests with colleagues in our research lab, I asked participants to create a model of a chair. I gave no instructions, except for explaining that they should experiment with mouse, the arrow keys and the page-up, page-down, insert and delete keys. A new user

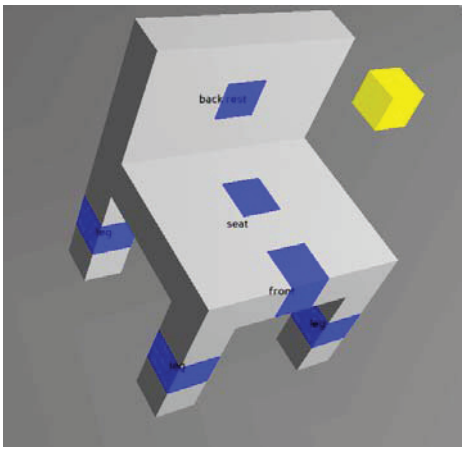

Figure 2. *The model of a chair: rotated and with some labels.*

typically begins by experimenting with the arrow keys (learning that the keys move the cursor). They then quickly discover the role of insert and delete. Within a minute, they have mastered the environment and begin to create a 3D model of a chair. This ease of use is in stark contrast to the hours (sometimes days) necessary to gain even basic competency in a 'professional' 3D modeling tool. In fact, reactions to the prototype were unexpectedly joyful. The simple interface reduces the barrier to creation and construction, acting as a playful tool especially for producing physical objects and other geometric patterns.

Role in Reasoning

The objective of the user-friendly interface and simplified representation scheme is to make it feasible to construct an enormous, reusable, large-scale knowledge-base of physical objects.

Because the knowledge will be collected from untrained volunteers, the resource will necessarily be informal, imprecise and contain redundancy. It will contain many different models of the same object: models at different levels of detail, with different shapes and with different labels. However, I do not view this informality as a weakness but as being of benefit to robust reasoning. For example, a system that has only one model of a prototypical chair might conclude that a chair always has four legs. In contrast, with many models of 'chair', the system would be able to deduce that chairs can have different numbers of legs and even that legs are a common, but not essential, property of chairs.

Consider the following knowledge that may be extracted from 3D models:

- The relative sizes of objects
- Whether and how one object can contain another
- Where objects might attach or connect to each other
- The general shapes and physical similarity of objects
- The spatial relationships of an object and its parts
- Where an object's 'affordances' might be said to be 'located' (*e.g.*, the top of a chair is used for sitting)
- Whether an object is likely to be stable if unsupported
- Whether an object may be used for other creative purposes (*e.g.*, using a chair to change a light-bulb)

Many of these questions can be answered by directly querying and manipulating the underlying representation. For example, one might use an Earth Mover's Distance metric over the voxels to measure the physical similarity of two objects. One might find the average position of voxels to estimate the center of mass.

Problems relating to containment, connection, attachment, stability and creative uses can be explored by instantiating the models in a physical simulation and observing the behavior. For example, one could test the stability of an unsupported object by instantiating a simulation and observing whether the simulated object topples over.

Note, however, that I do not claim that this scheme is *sufficient* for commonsense knowledge. No collection of objects described using this simple scheme will be able to answer nuanced questions about precise dynamics. Instead, the collection might be best viewed as a tool for generating 'first order' approximations of practical, spatial, commonsense reasoning problems. Given the difficulties and time associated with designing sophisticated theories of commonsense phenomena and situations, it may be the case that, for most domains of study, this simplified approximation will be the only approximation for the foreseeable future.

Conclusion

Sophisticated and deep theories of the commonsense world are important but, in a practical system, these need to be augmented with a broad knowledge base of factual data. In this paper, I have described a minimalistic but elegant scheme for representing knowledge: a scheme that I believe is the simplest solution that could be useful. This scheme is ideal for large-scale, crowd-sourcing of knowledge necessary for practical commonsense reasoning. While I have only reported on a prototype system, the tool will soon be published online and the models collected will be freely available.

References

Faaborg, A., Daher, W., Lieberman, H. and Espinosa, J. (2005) 'How to Wreck a Nice Beach You Sing Calm Incense', *Proceedings of the 2005 International Conference on Intelligent User Interfaces (IUI-05)*.

Goslin, M. and Mine, M.R. (2004) 'The Panda3D graphics engine', *Computer*, vol. 37, iss. 10, pp. 112–114.

Havasi, C., Speer, R. and Alonso, J. (2007) 'ConceptNet 3: a flexible, multilingual semantic network for common sense knowledge', *Proceedings of Recent Advances in Natural Language Processing (RANLP 07)*.

Johnston, B. and Williams, M-A. (2007) 'A Generic Framework for Approximate Simulation in Commonsense Reasoning Systems', *Proceedings of the International Symposium on Logical Formalizations of Commonsense Reasoning 2007*, AAAI Press, Palo Alto, California, pp. 71–76.

Shen, E., Lieberman, H. and Lam, F. (2007) 'What Am I Gonna Wear: Scenario-Oriented Recommendation', Proceedings of the 2007 International Conference on Intelligent User Interfaces (IUI-07).

Wood, L. (2005) 'Cycorp: the Cost of Common Sense', *Technology Review*, vol. 108, no. 3, pp. 33–33.