Building Domain Theories for Commonsense Reasoning from Language-Grounded Ontologies

Kenneth D. Forbus
Qualitative Reasoning Group, Computer Science Department, Northwestern University
2113 Tech Drive, Evanston, IL, 60208, USA
forbus@northwestern.edu

Abstract

One of the signature properties of commonsense reasoning is its breadth. Qualitative domain theories have been successfully built both by hand and by learning for small sets of phenomena, but scaling remains an issue. This paper describes an approach to achieving breadth by leveraging a large commonsense ontology. The idea is that a small set of concepts in the ontology corresponding to continuous processes and event types are identified, called *anchor concepts*. The subclasses of these anchor concepts form specializations of processes and types of events which then provide the desired breadth, e.g. that snowboarding is a form of motion. Pre-existing role relations for concepts of events and processes provide information about participants for QP model fragments and encapsulated histories. We show how this approach produces partial information about a broad range of continuous processes and event types. Rather than the usual carefully curated and bounded domain theories used in QR for modeling scientific and engineering reasoning, this approach to building domain theories is more open-textured. For example, the surface over which snowboarding occurs is usually snow and/or ice, something not currently stated in the ontology. The idea is that the rest of the specifications for any particular subclass of process will need to be completed by other means, e.g. instruction, experimentation, or hand-engineering.

Introduction

One of the original motivations for qualitative reasoning was to support commonsense reasoning. Even when the focus of QR is scientific or engineering reasoning, one of the jobs of qualitative models is to help in model formulation. Model formulation involves mapping from the unruly open everyday world to the tightly constrained formalisms often used in professional reasoning. For AI systems to be as helpful as a person in model formulation, they must have a reasonable understanding of the everyday world. Most qualitative domain theories have been generated by hand. Hand generation has been effective for many aspects of professional knowledge (e.g. aspects of physics, engineering thermodynamics, chemical engineering), but is daunting when considering the range of everyday phenomena. After all, people – with sensorimotor systems and learning abilities that are far more data-efficient than today's ML – take a decade or two to achieve broad commonsense knowledge, gleaned from a combination of direct experience and cultural inputs, including direct instruction. Progress has been made on building out knowledge bases using learning by reading, but most approaches require simplified text. Large language models should be useful in helping to expand the range of texts that can be processed. However, LLMs make poor knowledge bases for two reasons. First, their exposure to language is not grounded in the everyday world. Second, their success criterion is generating statistically plausible text, not correct reasoning. As the confabulation problems with LLMs show, these are at best only correlated. Hence we, like many others, continue to focus on using knowledge graphs as knowledge bases. Fortunately, there are now multiple large knowledge graphs such as Wikidata (Vrandecic & Krotzsch, 2014) that can provide broad knowledge (Forbus & Demel, 2022).

One of our hypotheses is that qualitative process theory captures aspects of natural language semantics (Forbus, 2019). One consequence of this hypothesis is that the underlying ontology in a commonsense knowledge base should in part reflect representational concerns relevant to qualitative reasoning. The everyday world includes many patterns of behavior that we can think of in terms of spatio-temporal units, the idea of *histories* introduced by Hayes (1985). Histories for objects are temporally extended but spatially bounded[1]. Histories are often defined in terms of the kinds of behavior happening in them. For example, one can think of filling a coffee cup or a swimming pool (or a basement). Filling can be accomplished by pouring

[1] By contrast, in the situation calculus, each situation is indeterminant temporally but spatially unbounded, which is a source of the frame problem.

from a pot in the case of a coffee cup, or pouring from buckets or a hose in the case of a swimming pool. These episodes are often delimited by qualitative changes in properties, e.g. for filling, the amount of fluid in the container being filled should be increasing during that episode. Histories can be hierarchical, e.g. filling a swimming pool using buckets will involve many filling/emptying of buckets, the emptyings of which all contribute to the filling of the swimming pool. The ability of qualitative representations to help segment perceptual information suggests that an important component of human commonsense knowledge is a broad vocabulary of descriptions of such types of events. Such events play a role in professional reasoning, since analyses are often couched in terms of them. Determining when to fire retro-rockets in a Mars lander, for example, requires conceptualizing the relevant part of its motions as a descent involving gravity, and solving for a firing time that will enable the lander to touch down safely. Event descriptions help provide boundary conditions, like the landing site and the desired speed on landing. *Encapsulated histories* in QP theory have been used to provide qualitative and quantitative models for such events that can be used in professional reasoning (e.g. Klenk & Forbus, 2009). Encapsulated histories can be learned via analogical generalization over descriptions of behaviors (Friedman & Forbus, 2008;2009). However, this has only been done for a small number of types of events.

Histories describe what is happening, but they do not explain why it is happening. QP theory introduced a notion of *continuous process* that provides a model for causal mechanisms in continuous domains. Pouring and filling in the examples above, for instance, would be explained in terms of a liquid flow process. The effects of such processes are compositional, so that models for specific systems can be formulated by combining them. Consider for example pouring water into a leaky bucket. There is a flow of water in, and a flow of water out – the intended flow and the leak are both explained in terms of the same type of continuous process. But whether or not the bucket is filling or not depends on the relative rates of the two flows. Thus the flows explain the filling episode. The everyday world contains many kinds of phenomena that we think of as continuous processes, such as motion, flows, phase changes, and so on. These general processes manifest in many ways. For example, motion can involve projectile motion through the air or empty space, moving along a surface, or various forms of water falling from the sky (e.g. rain, snow, hail). Hand-engineering model fragments for the full range of processes that manifest in our everyday world from scratch is daunting.

How can we leverage a broad commonsense ontology to build a commonsense QR domain theory? (Or, alternately, how to we bring the fruits of QR into efforts to ontologize commonsense knowledge?) Suppose we can identify within an ontology a set of high-level event types and processes that can serve as *anchor concepts* for a QP domain theory. That is, an anchor concept inherits from the concept of a type of encapsulated history or continuous process expressed in QP theory (e.g. motion), such that all of its more specialized concepts are aptly characterized by that domain theory construct. This provides a way of using the broad ontology to leverage well-engineered domain theory components. Moreover, if the ontology has mappings to natural language, then that ontology can be used in communicating with human partners, another requirement to achieve human-like model formulation.

This paper reports on work in progress exploring the use of a broad commonsense ontology to build a QP domain theory for commonsense reasoning. We start by summarizing the relevant background: aspects of QP theory and the NextKB knowledge base we are using. Then we discuss the issues involved in integrating QP theory with a broader domain theory, including processes versus events and continuous versus discrete levels of representations. A mapping of a small QP domain theory to NextKB is described next, demonstrating that this approach enables the range of phenomena that can be discussed to be considerably magnified. Finally, we discuss conclusions and future work.

Background

Qualitative process theory postulates *continuous processes* as the mechanisms for change in systems governed by continuous parameters. This model breaks down in some domains, e.g. analog electronics is better modeled by a component-centered ontology (de Kleer, 1984), and does not capture many of the spatial properties of motion (Forbus et al. 1991). Nevertheless, it appears applicable to a broad range of everyday phenomena. Recall that a QP domain theory consists of a set of schema, called *model fragments*, which can be instantiated to assemble models for particular scenarios and systems. Model fragments are specified by *participants* which indicate the kinds of entities it can be instantiated on, *conditions* which indicate when an instance of that model fragment is active, and *consequences* which are statements that hold for any time in which the conditions are true. Continuous processes are a subclass of model fragment that have *direct influences*, i.e. partial specifications of the derivative of some quantities of its participants, such that making a closed world assumption over the set of instantiated continuous processes specifies (qualitatively) the derivatives of those parameters.

As noted above, the consequences of processes hold at every instant within an interval over which that process is acting. To describe the cumulative effects of such processes requires histories for the objects affected, as per our example of filling a bucket earlier. To provide causal and mathematical constraints on episodes of histories, QP theory also provide a formalism for *encapsulated histories*, which can reference the temporal and spatial aspects of the episode they describe. These schemas are applied like model fragments, in that they have participants, conditions, and consequences. The consequences can be qualitative, e.g. the distance travelled in an episode of motion is qualitatively proportional to the time travelled. The consequences can also be quantitative, e.g. an equation describing distance travelled as a function of initial velocity and constant acceleration.

QP theory can be formalized in a variety of ways. Here we use an implementation grounded in the NextKB knowledge base, which is summarized below. This implementation has been used in several previous experiments and its details are not relevant for understanding this paper.

The NextKB knowledge base is an open-license resource being built at Northwestern University to support research in knowledge-rich AI and cognitive science. It builds on Cycorp's OpenCyc ontology, which provides a massive set of formally represented concepts and relationships. OpenCyc is an open-license subset of the Cyc ontology. Concepts are formally represented by *collections*, which can intuitively be considered as sets. For example, the collection Container represents all of the containers that there are, have been, will be, or might be. The relationship isa indicates that an individual can be considered an instance of that concept, e.g. (isa KenCollegeMug Container). There are inheritance relationships between concepts. The genls relation indicates inheritance between collections, e.g. (genls LiquidStorageTank Container) indicates that things which are storage tanks for liquids are also containers. There are also inheritance relationships among predicates, e.g. (genlPreds containerEntered toLocation) indicates that containerEntered implies toLocation holds between its arguments. The OpenCyc ontology is more expressive than most. For example, type-level predicates enable it to express higher-order statements, and modal operators (e.g. knows, beliefs) are included. This makes formalizing many concepts substantially easier than less-expressive ontologies. For example, (behaviorIncapable P1 SolvingAProblem thingAnalyzed) indicates that the problem P1 cannot be solved. There are many consistency constraints in the ontology. For example, disjointWith indicates that an instance of one collection cannot be a member of the other, e.g. (disjointWith Herbivore Carnivore). There are type constraints on arguments, arity, and the range of logical functions.

Some form of context mechanism is crucial for any representation system capable of considering alternative qualitative states, alternate perspectives in modeling (e.g. Falkenhainer & Forbus, 1991), or alternate domain theories. OpenCyc uses *microtheories* to provide a mechanism for contexts. Every fact holds in one or more microtheories. Microtheories inherit from each other via the genlMt relation. For example, (genlMt HumanSocialLifeMt HumanActivitiesMt) indicates that every fact believed in HumanActivitiesMt is also believed in HumanSocialLifeMt. Inheritance in all cases is monotonic. There are non-monotonic predicates to express dependence of some conclusions on the epistemic state of the system, e.g. believing something because one cannot infer its negation is a strategy that can be expressed and localized, rather than "wiring in" negation by failure as a global policy.

We distilled NextKB's ontology from the four available versions of OpenCyc. NextKB[2] includes over 82,000 collections, 26,000 relationships, 5,000 logical functions and 700,000 facts. We note that this is a small subset of the Cyc ontology, as found in the commercial version of Cyc and in ResearchCyc, both of which also have massively more axioms constraining the concepts and relationships in the ontology as well as a powerful reasoning engine that supports useful commonsense inferences complete with explanations based on dependency traces. For example, ResearchCyc can conclude that Earth cannot run a marathon, because no inanimate object can. We used the ResearchCyc knowledge base productively for a long time, but finally switched to OpenCyc to support dissemination and replication of our work.

In addition to OpenCyc contents, NextKB contains extensions for qualitative reasoning, including both QP theory and qualitative spatial reasoning, as well as visual/spatial capabilities used in CogSketch, our high-level vision system and sketch understanding system (Forbus et al. 2011; Forbus & Lovett 2021). Reasoning in these extensions is often conducted via procedural attachments to predicates, for efficiency. Analogical reasoning and learning is handled similarly. NextKB also has substantial natural language resources for English. It has a large lexicon, derived in part from a public-domain version of Webster's dictionary. Its semantics are organized using FrameNet frames, which have been mapped by hand to concepts in the OpenCyc ontology. FrameNet thus serves as a bridge between words and OpenCyc concepts. The lexicon has over 190,000 words and over 69,000 semantic translations. As noted above, AI assistants that help in model formulation need such broad language coverage, in order to communicate with their human partners.

[2] https://www.qrg.northwestern.edu/nextkb/index.html

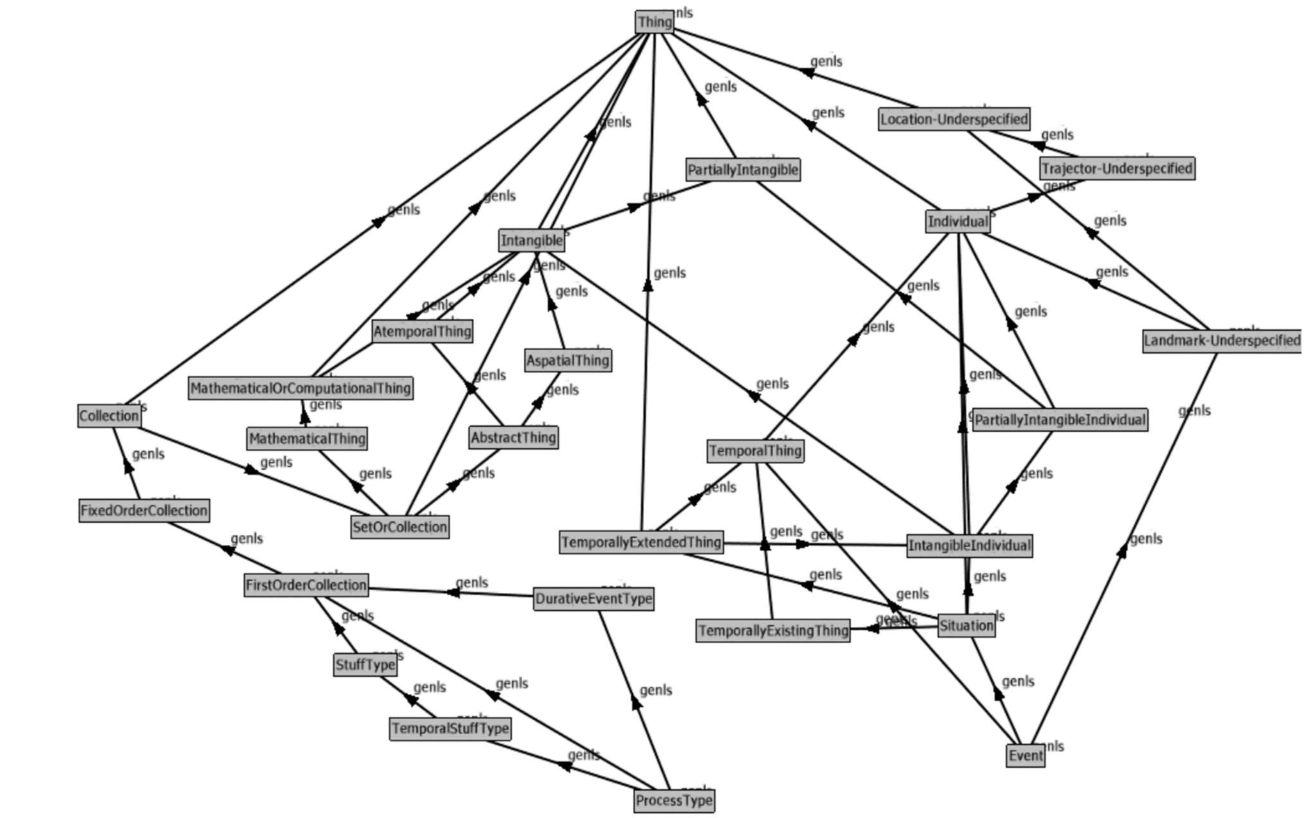

Figure 1: Partial view of the OpenCyc upper ontology showing where events and process types are grounded. It contains over 82,000 such concepts.

Ontological Grounding for Processes

All commonsense ontologies include some notion of event. Figure 1 shows how the general concepts of events and types of processes are related in the OpenCyc ontology. Generally there is a notion of sub-events, e.g. a wedding ceremony might include guests arriving, the exchange of vows, and merry-making. Processes are often represented in a similar way, with the difference being that the same properties are true of all of the sub-intervals within an occurrence of that process. This is compatible with the QP theory notion of a process being active whenever its conditions hold. Whether or not a phenomenon is treated as continuous or discrete depends on the granularity used in its description. A robust commonsense ontology must be able to support multiple levels of granularity, and OpenCyc does a reasonable job of this. For example, OpenCyc treats walking as a process, which is useful for estimating things like distance covered and effort expended. But it also provides support for describing the particular movements of legs up and down, discrete events within walking that are useful for purposes of physical therapy, for example. Another example is OpenCyc's concept of PrecipitationProcess, which is viewed as continuous, even though at a finer granularity, the movement of each raindrop or piece of hail can be viewed as a discrete event. Prior qualitative reasoning research has intermingled continuous and discrete perspectives in a similar way. For example, Rickel & Porter (1994) used time-scales in multiple perspective modeling of biological phenomena, given a particular time-scale of interest, their domain theory treated slower phenomena as exogenous constraints and quicker phenomena as functional connections.

Concepts describing processes form natural anchor points for QP-style continuous processes. That is, QP-style continuous processes are formalized as collections, and existing elements in the ontology inherit from them, thereby inheriting their schema. Figure 2 illustrates. However, not all commonsense processes are aptly described as continuous processes in the QP theory sense. For example, the concept of ProcessType in OpenCyc combines TemporalStuffType (thereby capturing the idea that the subintervals are the same) and DurativeEventType (thereby capturing the idea that occurrences of processes take time) and has 654 instances. Some of these are nicely expressed by QP theory, such as FluidFlow-Translation and PrecipitationProcess. Many others are not, including InternetSearching and IgnoringSomething. The difference is whether there are continuous parameters that aptly characterize the changes within an occurrence of a process. Uniformity in subintervals does not necessarily imply the existence of such parameters. Sometimes there are metaphorical extensions that can be applied. For example, an Internet search might be characterized in terms of progress towards the information-seeking goals for that search, or a decision-maker's thinking reaching a level of certainty about an action they are contemplating. We will not consider such metaphorical examples further here, but return to them in proposed future work below.

Linking QP continuous processes and encapsulated histories also requires linking the relationships that specify the participants for a model fragment. In English, for example, the subject of a motion verb indicates the object that is moving. The NextKB resources provide objectMoving as a relationship which formalizes this notion, enabling NLU systems to propose it as a possible meaning. Other spatial prepositions capture properties of an episode of motion. The spatial prepositions "from" and "to" can indicate the start and end of a motion, with "along" or "via" indicating its path. For example, From-TheWord has semantic translations that includes startOfPath (a spatial interpretation), intervalStartedBy (a temporal reading), and from-Generic (a more abstract version that includes the other two, but also the giver of a gift).

Analysis: Anchor Concepts in OpenCyc

To explore these ideas, we used pre-existing model fragments and encapsulated histories from QP domain theories for exploring the roles of qualitative reasoning in elementary school science tests (Crouse & Forbus, 2016), for learning textbook problem solving via cross-domain analogies (Klenk & Forbus, 2013), and some classic QP domain theories (Forbus, 1984). The goal is to estimate two properties: (1) How much leverage does the ontology provide us, in terms of additional phenomena covered? (2) Do the anchor

```
(in-microtheory PrecipitationQPMt)

(genlMt PrecipitationQPMt ScienceTestCollectorQPMt)
(genlMt ScienceTestInferenceQPMt PrecipitationQPMt)

;; model fragment definition
(isa NaivePrecipitationProcess QPProcessType)
(comment NaivePrecipitationProcess
  "Precipitation occurs when a liquid is in exposed to the air and its temperature is less than boil-
ing point but greater than its freezing point. The result of the process is that the liquid vaporizes
into an atmosphere.")

(mfTypeParticipant NaivePrecipitationProcess ?liquid LiquidTangibleThing liquidOf)
(mfTypeParticipant NaivePrecipitationProcess ?sub ChemicalCompoundTypeByChemicalSpecies substanceOf)
(mfTypeParticipant NaivePrecipitationProcess ?atmosphere GaseousTangibleThing atmosphereOf)
(mfTypeParticipantConstraint NaivePrecipitationProcess (substanceOfType ?liquid ?sub))
(mfiReverseConsequenceOf NaivePrecipitationProcess (and (isa ?rain RainProcess)
                                                        (products ?rain ?liquid)))
(mfTypeCondition NaivePrecipitationProcess (qGreaterThan
                                             (AmountOfFn ?sub Liquid-StateOfMatter ?atmosphere)
                                             SaturationPoint))
(mfTypeBiconditionalConsequence NaivePrecipitationProcess (hasQuantity ?self
                                                            (PrecipitationRateFn ?self)))
(mfTypeConsequence NaivePrecipitationProcess (qprop (PrecipitationRateFn ?self)
                                                    ((QPQuantityFn Temperature) ?liquid)))
(mfTypeConsequence NaivePrecipitationProcess (i+ (AmountOfFn ?sub Liquid-StateOfMatter ?liquid)
                                                 (PrecipitationRateFn ?self)))
(mfTypeConsequence NaivePrecipitationProcess (i- (AmountOfFn ?sub Gaseous-StateOfMatter ?atmosphere)
                                                 (PrecipitationRateFn ?self)))

;;; Anchor process
(genls PrecipitationProcess NaivePrecipitationProcess)
```

Figure 2: Example of a QP-style process anchored to the OpenCyc ontology

concepts provide connections to language that can be exploited by cognitive systems? To estimate leverage, we examine the subclasses of the anchor concepts. How many are there, and are they all reasonable? To estimate language coverage, we count the number of lexical items connected to the conceptual space covered by the anchor concept.

Table 1 shows the results for number of subclasses and words for reasonable anchor concepts for a set of pre-existing model fragments[3]. The anchor concepts were chosen to maximize applicability of the model fragment to the subclasses. This was straightforward for a number of model fragments, in particular, the basic processes involving fluids, heat, and phase changes. For example, the subclasses of liquid flow include DrinkingEvent and hence the words "drink", "imbibe", "quaff", "slurp" and "swill", among others. For heat flow, the subclasses include various forms of cooking (baking, barbecuing, steaming, roasting, and grilling). Not everything in the ontology is commonsense, e.g. the subclasses here include some ways that heating is used in semiconductor manufacturing, as well as global warming. This ability to expand to incorporate professional knowledge is a major advantage of starting with a broad ontology, and should simplify model formulation.

There are cases where the model fragments are somewhat too specific compared to the anchor process. Precipitation is an example: The model fragment concerns liquid leaving the atmosphere (as shown in Figure 2), whereas the PrecipitationProcess includes HailStormProcess, where what comes from the sky is ice. This could be resolved either by choosing a more specific subclass (e.g. RainProcess) or by slightly generalizing the model fragment. This issue comes up most strongly in motion, where there are general properties that hold (e.g. an episode in a motion history has a start, end, and velocity – motion that returns to its starting point is included) but also additional complications due to particular conditions, such as friction when sliding or gravity for projectiles. This has suggested ways to refactor our QP domain theories, i.e. to introduce encapsulated histories using purely qualitative mathematics for very abstract concepts of processes, to better capture the commonsense inferences that they license.

Motion is especially prolific. The 355 subclasses include things like snowboarding, flying by flapping wings, and parkour in addition to more traditional concerns of QR like projectile motion and sliding. It should be noted that in the ontology, PreciptationProcess entails motion, hence the words for that process and its specializations are a (small) subset of the words that refer to types of motion. Some of these subclasses have additional entailments over the basic QP model of motion, e.g. sliding entails the possibility of friction, and flying by flapping wings entails the use of energy supplied by the organism/artifact locomoting that way, as do walking and running. These additional distinctions could be captured by model fragments that elaborate motion, anchored to those concepts. For example, Flying-FlappingWings is a subclass of LocomotionProcess-Animal, so the common need for energy to accomplish locomotion, by whatever means, can be expressed once anchored on LocomotionProcess-Animal and also inherited.

We note that anchor concepts for some of the categories used in the participant constraints for model fragments are easily found, but others are not. An easy case is the general concept of container. The concept as used in these model fragments is reasonably captured by the collection Container, which has 2,787 subclasses and 1,757 words, although it includes many subclasses that someone might not usually think of in this way, e.g. dance clubs, airplane cabins, and a gigantic list of types of cars.

By contrast, it is difficult to find an anchor concept for the general concept of physical object (Physob, in classic QP domain theories). The closest is PartiallyTangible, which includes 42,339 subclasses, including things like butterflies and stores, but also concepts that are poor fits, such as the space under coffee tables. Similarly, concepts like thermal or volumetric objects, regularly used in compositional modeling for engineering domains, are not distinctions that the

Phenomena	Model Fragment Type	Anchor	Sub-classes	# Words
Liquid flow	LiquidFlowProcess	LiquidFlowEvent	26	15
Heat flow	HeatFlowProcess	HeatingProcess	44	50
Boiling	BoilingProcess	Boiling	5	3
Evaporation	Evaporation	Evaporation	0	1
Precipitation	NaivePreciptiationProcess	PreciptationProcess	14	21
Floating	ObjectFloatingInFluid	FloatingInASubstance	34	14
Motion	Motion	Movement-TranslationProcess	355	170
Friction	FrictionBetweenSolids	FrictionProcess	21	22

Table 1: Anchoring QP model fragments in NextKB

[3] We do not describe anchoring encapsulated histories to the OpenCyc ontology because our existing encapsulated histories, being developed later, were already integrated with OpenCyc because it is a subset of ResearchCyc.

OpenCyc ontology designers were concerned with. For such cases, it is straightforward to add the desired concepts to the ontology and incorporate subclasses of PartiallyTangible as appropriate. Moreover, such decisions can be incrementally learned from examples (Klenk et al. 2008).

So far we have looked at how much language coverage is added by anchoring QP constructs into the OpenCyc ontology. Are there words that are relevant to QP constructs that are not covered by anchor concepts? Yes. The exact number is hard to calculate, since it requires examining all of the lexicon. But, for example, the word "flow" uses FluidFlow-Translation, which includes both liquid and gas flow as subclasses. The QP models could be re-factored into a general fluid flow process with model fragments for liquids and gasses being model fragments specializing that one, or a system seeking to construct a qualitative model from a natural language description could gather candidate model fragments from subclasses of the mapped concept.

Conclusions and Future Work

The breadth of commonsense is a daunting challenge for qualitative reasoning. This paper argues that using a large-scale commonsense ontology (OpenCyc) that is tied to language (via NextKB) can help provide such breadth. The ability to find anchor concepts for model fragments and encapsulated histories from previous efforts is encouraging. The broad convergence in conceptual structure which makes FrameNet and OpenCyc mappable in the first place suggests that these commonalities are likely to be found in other resources, informed by the same cultural constraints. How this would vary given different cultures is a fascinating question. For example, how information is packaged into verbs varies across languages. In English one might say "The bottle floated into the cave" but in Spanish one would say the equivalent of "The bottle entered the cave, floating." Will those differences lead to cross-cultural differences in qualitative models?

We plan three lines of future work. First, we plan to refactor the QP model fragments and encapsulated histories to provide some of the intermediate representations that are currently missing, as well as use the ontology to help determine gaps where additional coverage is needed. Second, we plan to use this augmented domain theory to explore the construction of high-precision mental models during learning by reading, in order to learn new domain theory constructs and to solve problems expressed via language and sketching. Third, we plan to examine whether extending QP domain theories to more metaphorical uses supports inferences consistent with human metaphors (Lakoff & Johnson, 1981).

Acknowledgments

This research was supported by the Machine Learning, Reasoning, and Intelligence Program of the US Office of Naval Research, Grant N00014-23-1-2294.

References

Crouse, M. and Forbus, K. (2016). Elementary School Science as a Cognitive System Domain: How Much Qualitative Reasoning is Required? In *Proceedings of the Fourth Annual Conference on Advances in Cognitive Systems*. Evanston, IL.

de Kleer, J. (1984) How Circuits Work. *Artificial Intelligence*, **24**:205-280.

Falkenhainer, B. and Forbus, K. (1991). Compositional modeling: Finding the right model for the job. *Artificial Intelligence*, 51, 95-143.

Forbus, K. (1984). Qualitative process theory. *Artificial Intelligence*, 24, 85-168.

Forbus, K. (2019). *Qualitative Representations: How People Reason and Learn about the Continuous World*, MIT Press.

Forbus, K. & Demel, W. (2022). Integrating QR Quantity Representations with the Semantic Web. *Proceedings of QR 2022.*

Forbus, K., Nielsen, P., & Faltings, B. (1991). Qualitative Spatial Reasoning: The CLOCK project. *Artificial Intelligence*, **51**:417-471.

Forbus, K. & Lovett, A. (2021). Same/Different in Visual Reasoning. *Current Opinion in Behavioral Sciences*, 37:63-68, https://doi.org/10.1016/j.cobeha.2020.09.008

Forbus, K., Usher, J., Lovett, A., Lockwood, K., and Wetzel, J. (2011). CogSketch: Sketch understanding for Cognitive Science Research and for Education. *Topics in Cognitive Science*, 3(4), pp 648-666.

Friedman, S. and Forbus, K. (2008). Learning Causal Models via Progressive Alignment and Qualitative Modeling: A Simulation. *Proceedings of the 30th Annual Conference of the Cognitive Science Society (CogSci).* Washington, D.C.

Friedman, S. and Forbus, K. (2009). Learning Naïve Physics Models and Misconceptions. *Proceedings of the 31st Annual Conference of the Cognitive Science Society.* Amsterdam, Netherlands.

Hayes, P. (1985). Naïve Physics I: ontology for liquids. In J.R. Hobbs & R. Moore (Eds.), *Formal Theories of the Commonsense World*. Ablex: Norwood, NJ.

Klenk, M., Friedman, S., and Forbus, K. (2008). Learning Modeling Abstractions via Generalization. Proceedings of the 22nd International Workshop on Qualitative Reasoning. Boulder, CO.

Klenk, M. & Forbus, K. (2009). Analogical Model Formulation for AP Physics Problems. *Artificial Intelligence*, 173(18), 1615-1638. doi:10.1016/artint.2009.09.003

Lakoff, G., & Johnson, M. (1981) *Metaphors we live by*. University of Chicago Press.

Rickel, J. & Porter, B. (1994). Automated Modeling for Answering Prediction Questions: Selecting the Time Scale and System Boundary. *Proceedings of AAAI94*.

Vrandecic, D. & Krotzsch, M. (2014). Wikidata: A Free Collaborative Knowledgebase. *Communications of the ACM*, 57(10):78-85, October, doi:10.1145/2629489.