Neurosymbolic Qualitative Model Formulation from Natural Language: An Experiment

Kenneth D. Forbus
Qualitative Reasoning Group, Northwestern University
forbus@northwestern.edu

Abstract

One bottleneck for qualitative reasoning is translating from natural modalities such as language into formal models that support precise reasoning. This paper explores a hybrid combination of language models, which provide fluency, with symbolic natural language understanding, which provides high-precision semantics, to automatically build qualitative models from text. We use the QuaRel dataset to explore answering comparative analysis questions. Initial results from an implemented system are described.

1 Introduction

A signature strength of qualitative reasoning is its ability to derive conclusions from small amounts of incomplete information. Such inferences suffice for many tasks, or frame situations in ways that surfaces what issues might need to be addressed with additional knowledge. The generality of qualitative reasoning means it is broadly applicable in continuous domains, ranging from everyday physics to science and engineering tasks. The formality of qualitative representations, combined with its generality, raised the twin challenges of *model formulation* (Falkenhainer & Forbus, 1991) and *model interpretation*. Model formulation starts with descriptions of situations or systems in some approximation of everyday terms, and constructs a formal QR model aimed at supporting some task. Model interpretation is mapping the formal consequences derived from the model back into conclusions about the original system. Much work in the literature starts with some version of schematic or structural diagram. Here we go further afield and consider unrestricted natural language text as input. (We use an off-the-shelf dataset of comparative analysis problems developed by AI2, described below, thereby reducing tailorability.) Tackling this challenge requires two things. The first is finding ways to understand the rich variety of vocabulary, syntactic complexity, and irregularity of unrestricted text. We tackle this by a neurosymbolic approach, using statistical language models to provide fluency and symbolic NLU to provide high-precision understanding using a large-scale ontology. The second requirement is interfacing qualitative domain theories with natural language semantics at scale. Here we build on prior work (Forbus 2023), using structural relations in an ontology to enable reasoning to make such connections dynamically.

This paper describes work in progress. We start with providing the relevant background from language models, cognitive architecture, and qualitative reasoning. Then we describe our neurosymbolic approach to model formulation, including how LLMs are used to provide an initial framing of a problem, how events are analyzed using symbolic NLU assisted by an LLM for disambiguation, and how analogy is used to compare the models and solve the original problem. Experimental results are described next, focused on analyzing failure modes. Finally, some conclusions and plans for future work are discussed.

2 Background and Related Work

We use a subset of qualitative process theory (Forbus, 1984, 2018), with *encapsulated histories* as the main constituent of the domain theory. Histories are generated by the activity of processes, so encapsulated histories are schemas that can be instantiated to represent, for example, the occurrence of events. This is why they are used in textbook problem solving (e.g. Klenk & Forbus 2009), where reasoning about the start, end, and duration of events is common. The same is true of commonsense reasoning, e.g. the longer someone runs, the further they travel. Thus the usual machinery of quantities and fluents and qualitative proportionalities can be used with the temporal parameters of an encapsulated history.

2.2 LLMs and SLMs

One of the dramatic breakthroughs in AI recently has been the scaling up of pretrained language models. By exposure to truly massive amounts of text (and in some cases images and video as well), statistical models of language have been built that can be used for a number of tasks. Claims that such models can reason have proven controversial (e.g. Shojaee et al. 2025), and data contamination combined with lack of transparency about training materials makes evaluation of such claims difficult. Nevertheless, LLMs have been used as components in reasoning systems that involve language due to their broad coverage. In QR, for example, Suzuki & Yoshioka (2024) demonstrated that an LLM could be used to automatically construct model fragments for a harmonic motion example. Here we are asking less of LLMs. As described

below, we are using them in ways that stick to language, playing on their strengths.

While commercial LLMs as services are powerful, we prefer to use small open-source LMs (aka SLMs) that can be run locally. This has several advantages, including cost and replicability. Here we use Microsoft's Phi4 SLM (Abdin et al. 2024) because we have found it to be reliable. As a check we also used Gemini 3.5 Flash, with similar results.

2.3 Companion Cognitive Architecture

We build on the Companion cognitive architecture (Forbus & Hinrichs, 2017). Companions provide three essential services for this work. First, the Companion Natural Language Understanding system (CNLU; Tomai & Forbus, 2009) provides high-precision NL understanding that is used in constructing qualitative models for problems. Second, Companions provide NextKB,[1] a large-scale ontology based on OpenCyc and FrameNet, including a large-scale English lexicon and support for qualitative reasoning. Third, Companions provide analogical reasoning support, which is used in comparative analysis (Klenk et al. 2005).

2.4 Comparative Analysis Problems

Weld (1990) provided an elegant analysis of the kind of causal reasoning that compares properties of two systems, or alternate versions of the same system. A child rolling a toy car on a wooden floor might notice that it goes farther than when it is rolled on a carpet. Weld's formalization was restrictive in terms of the systems it applied to. Klenk et al. (2005) developed a more general method that constructed an analogical mapping between the systems being compared as a framework for ensuring causality worked the same way in them in relevant ways, e.g two different hand-drawn descriptions of systems. We use this method below.

There are two datasets of comparative analysis problems, both generated by AI2. QuaRel (Tafjord et al. 2018) consists of a large number of comparative analysis problems, all multiple choice, expressed in English text. The text was generated by crowdsourcing and does not seem to be regularized or otherwise cleaned up. The assumed background knowledge of 19 qualitative proportionalities was provided, so the challenge is to figure out which apply to solve the problems. Thus it is the breadth of types of situations, not the sophistication of the formal reasoning, being tested. QuaRTz (Tafjord et al. 2019) is like QuaRel in its problems, but provides 81 "grounding sentences" which express qualitative proportionalities textually. Here we use QuaRel, but we are also looking at QuaRTz (Xin & Forbus, 2026) as a knowledge capture problem.

3 Neurosymbolic Model Formulation

The general problem of model formulation involves starting with situations or systems described using natural modalities, such as language, sketches, and images, and constructing formal qualitative models that allow accurate and reliable reasoning, given some task. Our prior work on analogical model formulation has included comparative analysis problems with sketched input of physical systems (Klenk et al. 2005), but here we focus exclusively on text. The challenge is to work with "wild text", i.e. no limitations on domain, vocabulary, or syntax. We use this as a running example:

Example 1 (QuaRel_V1_Fr_0223):
Mike was snowboarding on the snow and hit a piece of ice. He went much faster on the ice because _____ is smoother.
(A) snow (B) ice

This is a nicely posed question. Generally crowdsourced text is full of spelling mistakes, grammatical errors, and conceptual oddities: In other words, representative of human language more generally. People are still mostly able to understand it. LMs can be used to extract information from them, so one approach might be to simply use an LM to generate formal qualitative models directly. This approach has several drawbacks. First, the ontology needed for commonsense is quite large. NextKB for example has over 100,000 predicates, making training a language model to produce representations in this ontology prohibitively expensive. Second, the process of model formulation needs to be inspectable and debuggable, since initial models may prove to be insufficient for the task at hand. By contrast, direct LLM to logic efforts use much smaller vocabularies (e.g. Noble et al. 2025; Raza & Milic-Frayling, 2025) and are more difficult to debug.

> "First, split this scenario into two analogous physical events, <DescriptionA> and <DescriptionB>. Each description should be a single natural language sentence. Each sentence needs to be kept as simple as possible, omitting extra words. Their grammatical form should be either <subject> <verb> or <subject> <verb> <preposition> <object> whenever possible. Use the same words as in the original text. Do not introduce pronouns or determiners. Avoid compound nouns, i.e. 'garden' instead of 'rock garden', unless the modifier is needed to tell <DescriptionA> and <DescriptionB> apart. Both sentences should be as alike as possible, because someone else will be comparing the events that they describe. Whenever possible, use the same type of event in each sentence. To output them, use EventA as the key for <DescriptionA> in the output JSON and use EventB as the key for <DescriptionB> in the output JSON."
>
> **Figure 1: Extracting events to compare**

In our approach we let LMs do what they do best: Make statistical judgments about language. This includes generating simplified English texts focused on what needs to be modeled, so that our symbolic NL understander (CNLU) can build a formal qualitative model.

The rest of this section outlines how our approach works, using a running example. We start with using an LM to frame

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

comparative analysis problems, then show how CNLU analyzes the events found by this framing, and finally how differential qualitative analysis is performed.

3.1 Problem Framing

The heart of a comparative analysis problem is understanding what is being compared. In Example 1, the comparison is between snowboarding on snow versus ice. An LM is used to produce two sentences, each describing one of the events to be compared. Here for example

- *EventA*: Mike snowboarding on snow
- *EventB*: Mike snowboarding on ice

Figure 1 shows the part of the prompt that produces this information. The restrictions on syntactic form of the event-describing sentences are to make them more tractable for CNLU. JSON output provides structure that makes it easier for output to be processed by the rest of the system.

> "Second, identify the type of quantity that the problem is asking about. Let's call it QueryQ, and it will be included in the JSON output using the key Query. QueryQ should be one of the following whenever possible: smoothness, friction, distance, speed, time, duration, acceleration, weight, strength, temperature, loudness, apparent size, size, brightness, sweating, fragility, flexibility, thickness.
> The JSON output will also include an entry whose key is QDir and whose value is as follows: If answer A implies that QueryQ is bigger in <DescriptionA>, then the value for the QDir entry in the output JSON will be -1. If answer B implies QueryQ is bigger in <DescriptionB>, then the value for the QDir entry in the output JSON will be 1. If you cannot tell which is correct, then the value for the QDir entry in the output JSON will be 0."
>
> **Figure 2: Identifying the sought quantity and interpretation of the multiple-choice options**

Next, the type of quantity being asked about is identified. In Example 1, the quantity type is smoothness, as per the last sentence of the problem and the options. One source of variation in the language of multiple-choice questions is the relationship between the choices and the rest of the problem. In Example 1, the choices are intended as possible fillers of an explicit blank. In others, the choices provide an ending to the incomplete last sentence of the problem. Yet another convention is the choices are a word or phrase that answers the explicit question raised by the last sentence. Handling this sort of variability is what LMs are good at, since they are trained on many multiple-choice tests. The trick is to produce a formal signal that will tell the rest of the system what to choose, based on the answer to its comparative analysis of the formal model. The LM is used to produce two pieces of information to handle this:

- *QuantityType*: The type of quantity the question is about.
- *QDir:* The relationship between the DQ value computed for the quantity type and the choices provided. If answer A implies that the quantity is bigger in EventA than EventB, then Qdir = -1, and 1 if the reverse.

QDir thus supports model interpretation, by encoding the LM's assessment of which choice suggests which output. In our running example, QuantityType is `SurfaceSmoothness` and QDir is `1`. Figure 2 shows the part of the prompt that estimates these properties. Note that there are two QuaRel dataset properties used in this prompt. The first is a list of the suggested quantities, based on the causal laws provided with the QuaRel dataset. This list was introduced to reduce variations in vocabulary produced by LMs. Even with the model temperature set to zero, LMs often vary vocabulary due to their training. The second is that QuaRel problems all have just two choices. By contrast, the Bennett Mechanical Comprehension Test (Klenk et al. 2005) has three, taking into account the possibility that the relevant quantity is the same. Comparative analysis can also be used in ranking problems, with even more choices, an extension worth exploring.

Type	Anchor Concept	Quantities
`TranslationEvent`	`Movement-TranslationEvent`	`Distance, Speed, Time`
`AcceleratedMotionEvent`	`Movement-TranslationEvent`	`Acceleration, Strength, Weight`
`TranslationOnSurfaceEvent`	`Motion-SolidAgainstSolid`	`Distance, Friction, Speed, SurfaceSmoothness, Temperature`
`HearingEvent (*)`	`Hearing`	`AcousticNoiseLevel, ApparentSize,Distance`
`SeeingEvent (*)`	`VisualPerception`	`ApparentSize, Distance, LightIntensity`
`Exercise (*)`	`Exercising`	`Duration, ExerciseIntensity, Sweating`
`GravityPull (*)`	`FallingEvent`	`Mass, Gravity`
`MaterialProperties (*)`	`ApplyingAPressure`	`Flexibility, Fragility, Strength, Thickness`

Table 1: Encapsulated histories, the concepts they are anchored to, and the quantities they constrain. A (*) indicates that the connection to language was not fully implemented due to lack of time.

Finally, any other quantity differences needed to solve the problem need to be identified. In Example 1, for instance, the difference in speed between the two situations (i.e. "faster on ice") provides the clue about a difference that can be propagated through a causal model to derive the answer. These quantities and their DQ values are generated by the portion of the prompt in Figure 3. Again, the list of QuaRel relevant parameters is included as a constraint. For our running example, the LM provides `"speed"` and `1` respectively.

"Third, identify what other quantities besides QueryQ are explicitly mentioned in the text. If they are not mentioned explicitly, do not include them. Do not include QueryQ. Whenever possible, they should be drawn from the following list: smoothness, friction, distance, speed, time, duration, acceleration, weight, strength, temperature, loudness, apparent size, size, brightness, sweating, fragility, flexibility, thickness. For each quantity explicitly mentioned, output the name of the quantity as the key of a JSON entry whose value is either -1 if that quantity is smaller in EventB than in EventA, 1 if that quantity is bigger in EventB than in EventA, and 0 if that quantity is the same in EventA and EventB. If you cannot tell its relative size across the two events, leave it out of the JSON output. Never use anything other than 1, -1, or zero as the value for a quantity."

Figure 3: Identifying other relevant quantity differences

Today's LMs are generally trained to be sycophantic user-facing chatbots. Thus prompts when trying to use them as components in a larger system commonly include admonishments to keep them on track. This includes asking them to not try to solve the problem themselves, not providing commentary in addition to the JSON output, and not festooning the JSON with extra material. Figure 4 illustrates those parts of our prompt.

Initial setup:
"You are a software component that does what it is told, carefully and accurately. I need you to do some intermediate work in breaking down a problem, so that it can be solved by different system. Here's the problem to be analyzed:"
<problem inserted here>
"DO NOT SOLVE THIS PROBLEM. Do not make assumptions that would answer the final question. All I want is something much simpler. Your output will be JSON. You must output ONLY valid JSON. Do not include code fences, backticks, comments, explanations, or any text outside the JSON object. Do NOT wrap the output in ```json or any other code fences. If you cannot express something in JSON, omit it. You must not say anything else, either before JSON being output or after it. You must not provide any other information other than what was requested. Do not comment or produce any other output besides what is specified here."

Figure 4: Overcoming LM chatbot training

3.2 Analyzing Events to Construct Models

As noted above, we use encapsulated histories to express domain knowledge. Thus the 19 qualitative proportionalities provided as part of the QuaRel dataset are collected into eight encapsulated histories, as shown in Table 1. Table 1 also shows the NextKB concepts they are anchored to, so that any concept which inherits from these concepts will also inherit the potential relevance of these encapsulated histories (Forbus 2023). For example, the concepts used in the semantics of verbs associated with sliding and rolling are `Sliding-Generic` and `RollingOnASurface` respectively. Both inherit from `Motion-SolidAgainstSolid`, which in turn inherits from `Movement-TranslationEvent`, so any instance of them will invoke these three encapsulated histories. There are many concepts that inherit from `Movement-TranslationEvent`, including `Snowboarding`, via `Sliding-Generic`. Thus the structural relations of the ontology, combined with a broad lexicon that maps words to concepts, enables systems to reach appropriate qualitative knowledge.

```
(def-encapsulated-history TranslationOnSurfaceEvent
    :participants ((theObject :type Physob
                    :constraints
                    (objectMoving ?self theObject))
          (theSurface :type PartiallyTangible
               :constraints
            (situationLocation ?self theSurface)))
  :conditions ((isa ?self Motion-SolidAgainstSolid))
  :consequences
  ((qprop- ((QPQuantityFn Speed) theObject)
           ((QPQuantityFn Friction) theSurface))
   (qprop- ((QPQuantityFn Distance) theObject)
           ((QPQuantityFn Friction) theSurface))
   (qprop ((QPQuantityFn Temperature) theObject)
          ((QPQuantityFn Friction) theSurface))
   (qprop- ((QPQuantityFn Friction) theSurface)
           ((QPQuantityFn SurfaceSmoothness)
                          theSurface)))
```

Figure 5: An example of an encapsulated history for the motion when two solids are in contact. QPQuantityFn coerces a Cyc-style value concept into a fluent compatible with QP theory

To instantiate an encapsulated history requires determining its participants from language. `TranslationOnSurfaceEvent`, shown in Figure 5, needs to find `theObject` (what's moving) and `theSurface` (what it is moving on). The constraints on these participants are chosen to be abstract relationships that inherit from the role relations used in the lexicon. For example, verbs of motion commonly use `eventOccursAt` to indicate where the event happened, which is a more specialized version of `situationLocation`[2]. Similarly, `objectMoving` is commonly used in motion events.

The assembly of a set of relevant encapsulated histories and finding their participants is carried out by reasoning based on CNLU's analysis of the LM-generated sentences for EventA and EventB. CNLU supports abductive reasoning for *narrative functions* that recognize task-relevant consequences implied by the semantics of CNLU constructs. Here

[2] Concepts inherit via the genls relationship, whereas predicates inherit via the genlPreds relationship. Both relationships are scoped by logical environments, which are constructed out of *microtheories*. Microtheories are used here to hold models, analyses of language, and to represent cases.

the narrative function rules look at whether there is an event which inherits from one of the anchor concepts, and if so, whether the appropriate participants can be found. The abductive mechanism is capable of making assumptions that resolve word sense and parse ambiguities in order to find relevant information. Additional disambiguation support is provided by using an SLM as follows. Given a set of alternate meanings for a word, symbolic NLG in CNLU constructs NL glosses for each of the alternatives. The SLM is then asked to pick the best option for the given context, which works surprisingly well (Zhao & Forbus, 2025). An SLM is also used to perform part-of-speech tagging, to constrain the alternate parses considered by CNLU.

The narrative function rules build statements specifying which encapsulated histories, if any, should be instantiated for the two events, including their bindings. The consequences of these encapsulated histories are combined into microtheories for each event. Figure 6 illustrates one of the two models constructed for our running example, the other is isomorphic up to discourse variables (e.g. `snowboard7927` which represents the snowboarding event) and ice instead of snow. This provides the final piece of the framing.

```
(isa snowboard7927 AcceleratedMotionEvent)
(isa snowboard7927 TranslationEvent)
(isa snowboard7927 TranslationOnSurfaceEvent)
(qprop ((QPQuantityFn Acceleration) mike7921)
       ((QPQuantityFn Strength) snowboard7927))
(qprop ((QPQuantityFn Distance) snowboard7927)
       ((QPQuantityFn Speed) snowboard7927))
(qprop ((QPQuantityFn Distance) snowboard7927)
       ((QPQuantityFn Time-Quantity) snowboard7927))
(qprop ((QPQuantityFn Temperature) mike7921)
       ((QPQuantityFn Friction) snow7962))
(qprop- ((QPQuantityFn Acceleration) mike7921)
  ((QPQuantityFn Weight) mike7921))
(qprop- ((QPQuantityFn Distance) mike7921)
  ((QPQuantityFn Friction) snow7962))
(qprop- ((QPQuantityFn Friction) snow7962)
  ((QPQuantityFn SurfaceSmoothness) snow7962))
(qprop- ((QPQuantityFn Speed) mike7921)
  ((QPQuantityFn Friction) snow7962))
(qprop- ((QPQuantityFn Time-Quantity) snowboard7927)
  ((QPQuantityFn Speed) snowboard7927))
```

Figure 6: Qualitative model for "Mike snowboarded on snow."

3.3 Performing Differential Qualitative Analysis

Differential qualitative analysis begins with constructing an analogy between the models of the two events. This is done via the Structure-Mapping Engine (SME; Forbus et al. 2017). SME builds one or more *mappings* that include correspondences between expressions and entities in the two descriptions and a score indicating the quality of the mapping. The best mapping between the two models provides the alignment needed to solve comparative analysis problems. We define

- `(dqValue <q> <m> <1,0,-1,ambig>)` the differential qualitative value of quantity `<q>` relative to mapping `<m>`, which is either increased (`1`), constant (`0`), decreased (`-1`) or ambiguous (`ambig`).

There are several external ways of computing DQ values. The system is capable of using numerical values or visual quantities, for example. For QuaRel the other values deemed relevant are converted into ordinals for corresponding quantities across the comparison, e.g. in Example 1, the value of 1 for speed indicates that the speed in EventB is higher than the speed in EventA. This information is propagated through the qualitative proportionalities in the two models to derive a DQ value for the target quantity. That DQ value is then combined with the QDir computed for the target quantity to tell the system which answer choice to make.

4 Experimental Results

We use the QuaRel development set for this experiment. It consists of 278 problems, which can be characterized as Exercise (16), Friction (140), Gravity (8), Hearing (13), Material (20), Motion (39), Seeing (22), and Strength (21). Given that time did not permit a full implementation of the bridge between language and encapsulated histories, we report only on Motion and Friction problems. The system answered correctly 6 Motion problems (13%) and 18 Friction problems (13%). The system does not guess. It answers incorrectly 10% of the time for Motion problems and 16% of the time for Friction problems.

The system was instrumented to gather data about when in the problem-solving process failures were occurring. Knowing when a problem occurs does not always uniquely determine the source of the problem. Here is our failure taxonomy and the percentage of such failures across all the problem topics in the development set:

- Framing (7.5%): The LLM failed to produce one or more of the problem framing elements (Section 3.1).
- Modeling (48%): One or both of EventA/EventB have no encapsulated histories. This is due either to problems in the EventA/EventB sentences generated by the LLM or CNLU's interpretation of them.
- Solve (34%): Differential qualitative analysis fails, either because of errors in choice of other parameter and change by LLM or incomplete EH models.
- Wrong Choice (10.6%): Incorrect answer was chosen, due to either a mistake in query quantity or Qdir (Section 3.1)

The bridge between language and qualitative models provided by this approach is not terribly robust. Let us examine some representative examples in more detail to get a sense of how it might be improved.

4.1 Qualitative Analysis of Failures

In any crowdsourced dataset there can be problems that are simply out of scope, e.g.

QuaRel_V1_B5_0842: Lebron James a strong player for the Cavs battles Kevin Durant a thin player for a rebound. Who is likely to get the rebound? (A) Durant (B) Lebron

These are thankfully rare. A more common source of problems is automatic disambiguation making suboptimal choices. Consider

QuaRel_V1_B5_0583: Jen is at the gym exercising. Jake is at home sleeping. Because of this (A) Jen is sweating more (B) Jake is sweating more.

Phi4 selected a meaning for "exercising" that was `UsingAnObject`, which is more about skilled performance with an implement than physical exercise (`Exercising`). Similarly, in

QuaRel_V1_Fr_1360: Tom dropped his lead pipe on the sidewalk and had to chased after it a long way before it stopped. Tom entered his yard and dropped his lead pipe again this time in the grass but noticed it didn't go very far at all. Tom realized that the lead pipe received far greater resistance when dropped on (A) Grass (B) Sidewalk

EventA: Tom dropped lead pipe on sidewalk
EventB: Tom dropped lead pipe on grass

Phi4 selected some bizarre meanings: "lead" as per the FrameNet frame `Criminal_Investigation`, "pipe" as `Pipe-SmokingDevice` and "grass" as `Marijuana`. One might argue that these choices are statistically internally coherent, and providing more of the original problem might help. Interestingly, Gemini on this problem also selected `Pipe-SmokingDevice`, but `Grass-Plant` for "grass", and its version of the events left out "lead", thereby avoiding that issue.

The deeper problem that these two examples also share is that the common events are not cast in ways to bridge to qualitative domain model anchors. In QuaRel_V1_B5_0583 the system was one disambiguation mistake away from finding a relevant qualitative model for exercising, but not so for sleeping. The QuaRel domain theory is missing an intermediate parameter: level of metabolic activity is a causal intermediary between exercise intensity and sweating. The direct connection in QuaRel is a simplification, a reasonable one in many circumstances, but not even for their own problems. This suggests that the ability to modulate model complexity during model formulation can be as important for commonsense as it is for engineering problem solving (Falkenhainer & Forbus, 1991).

In QuaRel_V1_Fr_1360 the problem is more subtle: The real event being reasoned about is the pipe *rolling*, which is not stated explicitly anywhere. Language elides that which is common, including well known types of translations in qualitative states. We know from experience that pipes are often cylinders, and that when dropped, cylinders typically roll. Neither Phi4 nor Gemini detected this. Could they? Perhaps, with an appropriate prompt, but introducing new constructs also widens the door to hallucinations. On the other hand, some schematic qualitative simulation, using type-level reasoning (e.g. in NextKB, `(typeBehaviorCapable Cylinder RollingOnASurface objectMoving)`) or analogical chaining (Blass & Forbus, 2017) offer more inspectable and controllable options for filling in commonsense implications of language.

More broadly, a system's conceptual knowledge should be used to help guide the construction of mental models of situations from language. Consider

QuaRel_V1_Fr_1137: A train travels at a greater rate of speed while on tracks than while rolling over gravel. This means that the _____ creates more friction (A) tracks (B) gravel.

Phi4 produces events that prioritize original language at the cost of degrading the comparison:

EventA: Train travels on tracks
EventB: Train rolls over gravel

"rolls" of course brings up three encapsulated histories, while "travels" brings up none, so using Phi4, the system was missing a model. Gemini used "travels" for both events and using it for disambiguation led to the selection of "travels" as an instance of `Travel-TripEvent` with the `travelerInTrip` being the train. Again, commonsense should play a role: When trains travel normally, they do so by rolling. (Sliding can occur when brakes are applied and flying occurring means something is terribly wrong, of course.) A failure to derive a qualitative model for "travels" that leaves out differences in the two situations being analyzed suggests retrieving a model for how trains move that has a causal dependency on the surface materials. World knowledge organized by the event structure of language woven together by qualitative causal relationships provides the missing components for solving these sorts of problems, and perhaps many others.

Finally, several examples in QuaRel suggest that our current notion of anchor concept which always connects encapsulated histories to events is too restrictive. Gravity for example is not an event, and comparison problems involving it sometimes do not mention events at all. Learning all of the places where concepts are relevant might be done incrementally by analogical model formulation (Klenk & Forbus, 2009), for example.

5 Conclusions and Future Work

The difficulty of model formation and interpretation stems in part from its requirement to work with natural modalities and in part from the broad commonsense knowledge needed to accurately apply qualitative domain theories to the everyday world. The breadth of coverage provided by today's language models now enables some processing to be done even with "wild" text. But they are a long way from a panacea. This paper represents another step down the hybridization road, a neurosymbolic approach to language that seeks to combine the fluency and coverage of statistical methods with the inspectability and debuggability of symbolic representations. Some lessons so far:

1. A greedy approach to model formulation from language, where LMs are assumed to be sufficiently optimal to give workable answers, is insufficient. Backtracking and search is needed.
2. The breadth of coverage in a large lexicon and inheritance in a large conceptual ontology is necessary but not sufficient to bridge between language and commonsense knowledge. More knowledge of the everyday world would enable systems to fill in the gaps left in language. A combination of examples, generalizations from examples, and type-level laws distilled from generalizations, might provide sufficient grounding for this aspect of commonsense.
3. The scale of knowledge to be acquired makes data-efficient incremental learning methods crucial. While pretrained LMs are useful for some tasks, for building trustable collaborative AI systems, we need systems that learn at human rates and whose reasoning and knowledge is sufficiently inspectable to be trusted.

Our next steps are to finish the language/QR bridge for the QuaRel and QuaRTz domains, and implement search strategies for more reasoning during model formulation. This reasoning needs to include applying commonsense inferences (e.g. typical behaviors, either via type-level first principles reasoning or via analogical reasoning) to construct coherent mental models of the situations described by language. Then we plan to broaden the range of qualitative models and reasoning supported, beyond comparative analysis, using science tests as a means of evaluation.

Ethical Statement

There are no ethical issues.

Acknowledgments

This research was supported by the US Office of Naval Research. Wangcheng Xu and Zoie Zhao helped with LLM APIs and disambiguation, respectively.

References

Abdin, M. et al. 2024. Phi-4 Technical Report. https://arxiv.org/abs/2412.08905

Blass, J. & Forbus, K. (2017). Analogical Chaining with Natural Language Instruction for Commonsense Reasoning. *Proceedings of AAAI 2017*.

Crouse, M., McFate, C.J., & Forbus, K. (2018b). Learning to Build Qualitative Scenario Models from Natural Langauge. *Proceedings of QR 2018*, Stockholm.

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. D., Ferguson, R. W., Lovett, A., and Gentner, D. (2017). Extending SME to handle large-scale cognitive modeling. Cognitive Science, DOI: 10.1111/cogs.12377, pp 1-50.

Forbus, K. (2018). *Qualitative Representations: How People Reason and Learn about the Continuous World*, MIT Press.

Forbus, K.D. & Hinrichs, T. (2017) Analogy and Qualitative Representations in the Companion Cognitive Architecture. *AI Magazine*.

Forbus, K. (2023). Domain Theories for Commonsense Reasoning from Language-Grounded Ontologies. *Proceedings of QR 2023*, Krakow, Poland.

Klenk, M., Forbus, K., Tomai, E., Kim,H., and Kyckelhahn, B. (2005). Solving Everyday Physical Reasoning Problems by Analogy using Sketches. *Proceedings of 20th National Conference on Artificial Intelligence* (AAAI-05), Pittsburgh, PA.

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

La Malfa, E. Petrov, A., Frieder, S., Weinhuber, C., Burnell, R., Nazar, R., Cohn, A., Shadbolt, N., & Wooldridge, M. (2025). Language-Models-as-a-Service: Overview of a New Paradigm and its Challenges. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(27), 28742-28742

Noble, B., Blanck, R., & Wijnholds, G. (2025) In the Mood for Inference: Logic-based Natural Language Inference with Large Language Models. *Proceedings 5th Workshop on Natural Logic Meets Machine Learning*, ACL.

Raza, M. & Milic-Frayling, N. (2025) Instantiation-based Formalization of Logical Reasoning Tasks using Language Models and Logical Solvers. *Proceedings of IJCAI-2025*.

Ribeiro, D. & Forbus, K. (2021). Combining Analogy with Language Models for Knowledge Extraction, *Proceedings of the Third Conference on Automatic Knowledge Base Construction*.

Shojaee, P., Mirzadeh, I., Alizadeh, K., Horton, M., Bengio, S. & Farajtabar, M. (2025) The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity. https://ml-site.cdn-apple.com/papers/the-illusion-of-thinking.pdf

Suzuki, S., & Yoshioka, M. (2024) Preliminary Experiments of Qualitative Reasoning Model Construction Using Large Language Model, *Proceedings of QR2024*, Santiago de Compostela, Spain.

Tafjord, O., Clark, P., Gardner, M., Yih, W. & Sabharwal, A. (2018) QuaRel: A Dataset and Models for Answering Questions about Qualitative Relationships, *AAAI-2018*.

Tafjord, O., Gardner, M., Lin, K. & Clark, P. (2019) QuaRTz: An open-domain dataset of qualitative relationship questions. *EMNLP 2019*.

Tomai, E. and Forbus, K. (2009). EA NLU: Practical Language Understanding for Cognitive Modeling. *Proceedings of the 22nd International Florida Artificial Intelligence Research Society Conference.* Sanibel Island, Florida.

Weld, D. (1990). *Theories of Comparative Analysis*. MIT Press.

Zhao, K. & Forbus, K. (2025). Integrating Symbolic Natural Language Understanding and Language Models for Word Sense Disambiguation. Proceedings of Advances in Cognitive Systems 2025.