Learning to Build Qualitative Scenario Models From Natural Language

Maxwell Crouse, Clifton McFate, & Kenneth Forbus
Qualitative Reasoning Group, Northwestern University
2133 Sheridan Road, Evanston, IL, 60208, USA
mvcrouse@u.northwestern.edu, {c-mcfate, forbus}@northwestern.edu

Abstract

Natural language descriptions are commonly used to communicate situations where qualitative reasoning is relevant, but automatic construction of scenario models from language remains a challenging problem. This paper describes an approach for learning how to construct scenario models from natural language text, using small amounts of language-only training data. Evidence Expressions (EEs), which bridge between the general-purpose outputs of a semantic parser and the formal constructs of a qualitative domain theory, are learned from training examples. These EEs are then retrieved and combined by analogy to create scenarios for more complex problems. We demonstrate the effectiveness of our technique on a set of 4th and 5th grade science test questions.

1 Introduction

Understanding how to extract qualitative models in the process of natural language understanding is an important problem for learning by reading (e.g. [Kuehne & Forbus, 2004; McFate et al. 2016b]) and for using that knowledge in answering questions. For example, Crouse & Forbus [2016] found that 29% of elementary science test questions they examined required using qualitative reasoning to answer. This suggests that shallow understanding systems (e.g. [Khashabi *et al*, 2016; Clark *et al*, 2016; Khot *et al*, 2017]) will not suffice for human-level performance on such tests. Given the breadth of natural language, learning approaches look like the only option.

Learning in this domain is difficult: There do not exist the large annotated corpora needed for modern machine learning methods. Even if there were, such approaches lead to building domain-specific language systems, which do not gracefully extend to other domains. Since many interesting problems combine multiple domains, generality is also an important constraint. Our approach [Crouse *et al*, 2018] is to use a domain-general semantic parser which is extended by learning abductive patterns that connect the general-purpose semantics to the representations needed for reasoning about various domains. These *Evidence Expressions* (EEs) are then applied to new problems via analogy. Importantly, EEs are compositional, in that multiple EEs learned for simpler texts can be combined (via multiple analogies) to provide interpretations for more complex texts. In our prior work, EEs were constructed by connection graph techniques, using as input unannotated natural language question-answer pairs (the classic Geoquery factoid Q/A domain). This paper extends these ideas to handle more complex, multi-sentence training examples, as needed for building scenario models.

We begin by reviewing the relevant background. Then we describe our technique, including both learning and application of the learned knowledge. We evaluate our approach by answering questions requiring model formulation from a set of 4th and 5th grade science questions.

2 Background

2.1 Qualitative Process Theory

Qualitative Process Theory [Forbus, 1984] formalizes processes as the mechanism underlying continuous change. The direct effects of a process (e.g. liquid flow into a tub) are called *direct influences* (represented as i+ and i-), and their indirect effects are called *indirect influences* (represented as qprop/qprop-), e.g. the level of the water in the tub. *Model fragments* are compositional schemas that define types of entities and relationships in the world. They have *participants* which are related by the model fragment, *constraints* among participants that determine when the model should be considered, and *conditions* of activation. When a model fragment is active, its *consequences* hold. These are frequently influences, though other relationships can be consequences as well. Consider for example the model fragment in Figure 1, which describes a contained liquid. The first line defines its name. Lines 2 and 3 define the participants, entities of types `Container` and `ContainedStuff`. They play the role of `containerOf` and `containedObject` respectively in an instantiated model fragment. Line 4 defines a constraint, that this model should

```
1. (isa SimpleContainer ContainedSubstance)
2. (mfTypeParticipant ContainedSubstance
         ?container Container containerOf)
3. (mfTypeParticipant ContainedSubstance
        ?stuff ContainedStuff objectContained)
4. (mfTypeParticipantConstraint SimpleContainer
        (PhaseOf ?stuff Liquid)
5. (mfTypeConsequence SimpleContainer
       (qprop ((QPQuantityFn Pressure) ?stuff)
              (AmountFn ?stuff)))
```

Figure 1. *A simple qualitative model of a contained liquid*

only be considered for liquids. Finally, Line 5 provides a consequence, that an indirect influence holds between the pressure and amount of contained stuff.

Given a domain theory consisting of a set of model fragments and a scenario description, a *model formulation* algorithm instantiates model fragments based on which preconditions are met by the scenario. In traditional QR, programs that provide structural descriptions do so in the ontology of the domain theory. Here, the challenge is to learn connections between everyday concepts (e.g. tub) and concepts in the domain theory ontology (e.g. container).

2.2 Semantic Parsing

We use the Explanation Agent NLU (EA NLU) semantic parser [Tomai & Forbus, 2009]. EA NLU is a bottom-up rule-based chart parser that uses a feature based grammar. It uses the NULEX lexicon [McFate & Forbus, 2011] and Fillmore et al's [2001] FrameNet. FrameNet ties words to a semantic schema and annotates how semantic roles are bound to arguments in syntactic patterns.

As an example, the word *change* evokes the `Cause_change` frame which has semantic roles that include `Initial_category`, and `Final_category`. When used in the sentence "The snow changes to water.", the first noun phrase is the `Initial_category`, and the prepositional phrase is the `Final_category`. These patterns of role bindings (called valence patterns) are stored in the ontology in templates that get bound in the grammar. In our system, both the grammar and semantic templates are represented in the Cyc ontology [Matuszek *et al*, 2006]. Semantic ambiguities are represented via mutually exclusive choice sets, e.g., the word *ball* in a sentence would lead to including a choice set for a toy versus a dance in the parser's interpretation. Our system operates over these choices to reason about which combination of them would lead to a good qualitative model.

2.3 Analogy

We use the Structure Mapping Engine [Forbus *et al*, 2017], a computational implementation of Gentner's [1983] Structure Mapping Theory. Structure mapping aligns hierarchical structured representations (predicate calculus) according to the principles of SMT, that each element in a case may match with at most one in the other and that the children of matched elements also match. After alignment, structure in one case that is missing from the other can be inferred by analogy (*candidate inferences*).

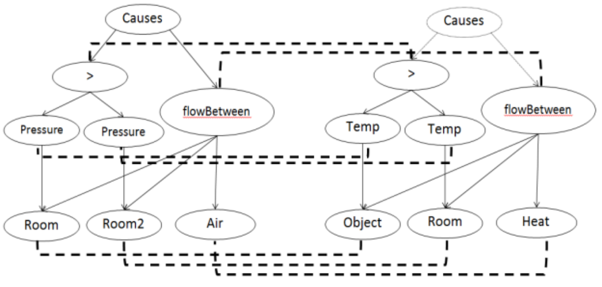

Figure 2. *An example SME mapping*

Consider Figure 2 which shows a mapping between a model of heat flow and a model of air flow. In the base (left), a difference in air pressure between two rooms causes air to flow from one to the other. In the target (right), a difference in temperature exists between a hot brick and cool room. As in the base, there is a flow process between the two entities. When aligned, the ordinal relations and flow process match, allowing the inference that, as in the case of air flow, a quantity difference drives heat flow.

We use the MAC/FAC model of analogical retrieval [Forbus *et al*, 1995] and the SAGE model of analogical generalization [McLure *et al*, 2015]. MAC/FAC is a two-phase model of analogical retrieval that uses a cheap preliminary feature-vector match to generate candidates for its second stage, which uses SME to select the most similar retrieval. SAGE incrementally accumulates examples in a *generalization pool*, a kind of case library, which contains both examples and automatically constructed generalizations. When a new example arrives, it uses MAC/FAC to find the most similar item, and assimilates them if they are sufficiently similar. The assimilation process produces (or updates) a generalization, where the probability of each statement in it depends on the number of examples that contribute corresponding statements. In Figure 2, for instance, the flow-between statement would have probability 1, whereas the temperature/pressure statements would have probability 0.5.

3 Our Technique

Following Crouse *et al* [2018], we represent the mapping between lexical semantics and model fragment ontology as *Evidence Expressions*. An evidenceForExpression statement (EE) can be viewed as an abductive rule, where the antecedents are semantic choices and the consequents are a task-specific logical form. In this case, consequents are the participant relations and conditions necessary to instantiate a set of model fragments.

```
1.  (isa ?self NaiveMeltingProcess)
2.  (mfTypeParticipant NaiveMeltingProcess ?thing-melting
      SolidTangibleThing focusOf)
3.  (mfTypeParticipant NaiveMeltingProcess ?sub
     ChemicalCompoundTypeByChemicalSpecies substanceOf)
4.  (mfTypeParticipantConstraint NaiveMeltingProcess
      (substanceOfType ?thing-melting ?sub))
5.  (mfTypeParticipantConstraint NaiveMeltingProcess
      (relationAllInstance freezingPoint ?sub ?m-temp))
6.  (mfTypeCondition NaiveMeltingProcess (qGreaterThan
     (TemperatureFn ?thing-melting) ?m-temp))
7.  (mfTypeConsequence NaiveMeltingProcess (qGreaterThan
     (LiquidGenerationRateFn ?self) 0))
8.  (mfTypeConsequence NaiveMeltingProcess (qprop
     (LiquidGenerationRateFn ?self) (TemperatureFn ?thing-melting)))
9. (mfTypeConsequence NaiveMeltingProcess (i- (AmountOfFn ?sub
     Solid-StateOfMatter ?thing-melting)(LiquidGenerationRateFn ?self)))
10. (mfTypeConsequence NaiveMeltingProcess
     (i+ (AmountOfFn ?sub  Liquid-StateOfMatter ?thing-melting)
        (LiquidGenerationRateFn ?self)))
```

Figure 3. *A qualitative model of melting*

EEs learned during training can be applied to novel questions via analogy. More concretely, question semantics are aligned to the antecedents of an EE retrieved via MAC/FAC, and the consequent of the EE is then instantiated with the entities of the novel question via analogical inference. Missing antecedents are allowed at a cost and a recombination algorithm determines the smallest set of EEs whose antecedents cover the entirety of the question semantics [Crouse *et al*, 2018].

An example will make this clearer. Figure 3 shows a simple model for melting. Informally, the requirements for activation (lines 2-6) are that there is a solid (line 2), and that its temperature is greater than its melting point (line 5 and 6). Now consider the question described in Figure 4. That packed snow is a solid is not explicitly stated, nor is the initial ordinal relationship of its temperature to the melting point of water. That the room is warm is relevant, since that could lead to a heat flow, and thus a melting. That is consistent with the snow changing to liquid. Finally, there are, from the standpoint of model formulation, irrelevancies, e.g. that a student is conducting an investigation, although the savvy student would read "changes in the state of matter" as a hint. We use this example to illustrate how our technique works.

3.1 Training

Our technique takes as inputs a natural-language scenario paired with the active process of the scenario, drawn from science test questions. For the problem in Figure 4, it would be given the text paired with the word "melting" which refers to the `NaiveMeltingProcess` model fragment. This results in an initial set of model fragments consisting of `NaiveMeltingProcess`, any model fragment participants, as well as any fragments that they depend on. This retrieved set will be referred to as our *target set of expressions*.

A student is investigating changes in the states of matter. The student fills a graduated cylinder with 50 milliliters of packed snow. The graduated cylinder has a mass of 50 grams when empty and 95 grams when filled with the snow. The packed snow changes to liquid water when the snow is put in a warm room. Which statement best describes this process?

Figure 4. *An example question involving "melting"*

EA NLU interprets the scenario. Figure 5 shows a partial set of EA choices for the text in Figure 4. In Figure 5, each word has its alternative predicate calculus interpretations listed as sub-bullet points. The first step is to map relevant elements of the natural language scenario to models and their conditions. This proceeds in three phases: The textual semantics and model are aligned in an *initial matching*. Relevant aspects of the semantics are also found through *stability analysis*, and finally *Steiner tree connecting semantics* are found to bridge disjointed antecedents. Figure 6 illustrates this process.

3.1.1 Initial Matching

Let T be the target set of logical expressions (model fragments) and S be the set of semantic choices (e.g. 1a, 2a-c, 3a-b, and 4a in Figure 5). The initial alignment step operates over the complete bipartite graph $G = (S, T, S \times T)$. A conflict-free matching M in G is a set of vertex-disjoint edges (one-to-one correspondence) such that no two pairs of edges can have expressions from T that conflict in C. In other words, if there is a conflict pair $(t_k, t_l) \in C$ then M cannot simultaneously contain a pair of edges like (s_i, t_k) and (s_k, t_l).

This negative-disjointness constraint makes the matching problem NP-Hard [Darmann *et al*, 2011]. For efficiency, we use a local search procedure that starts from a promising

1. "graduated cylinder"
    1a. (isa graduated-cylinder751136 GraduatedCylinder)
2. "warm"
    2a. (ambientTemperature room752597 Warm)
    2b. (temperatureOfObject room752597 Warm)
    2c. (temperatureOfObject room752597 Warm)
        – 2b and 2c are justified by different parse trees
3. "water"
    3a. (isa water751940 (LiquidFn Water))
    3b. (isa water751940 Water)
4. "snow"
    4a. (isa snow751995 SnowMob)

Figure 5. *A subset of the semantics produced for our example question. 2b and 2c shows a situation where the same semantics are justified by different parse trees.*

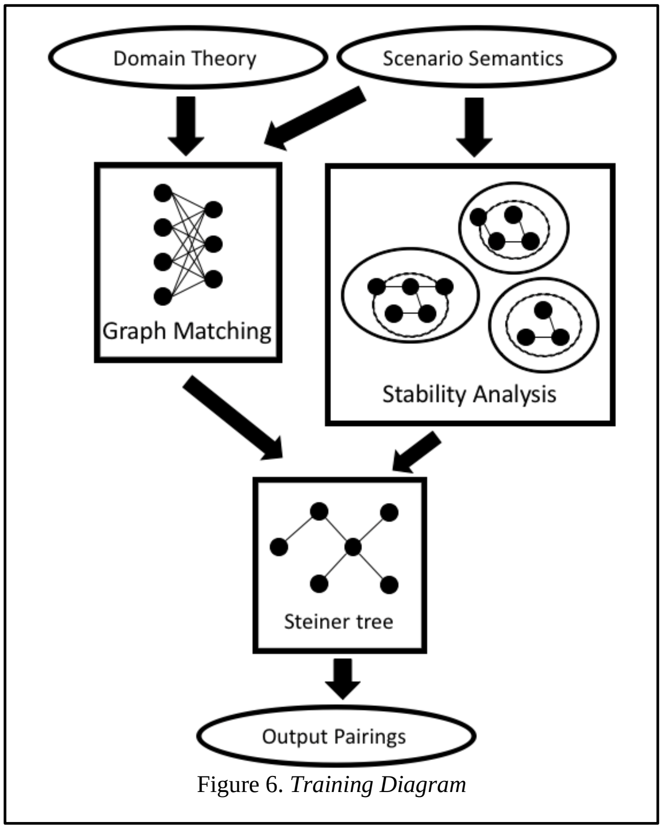

Figure 6. *Training Diagram*

candidate solution and moves amongst better neighboring solutions until it can no longer improve. For a conflict-free matching *M* in *G*, it expands outwards to all conflict-free matchings that differ by at most two edges from *M*.

The score of a conflict-free matching *M* is determined by three properties: ontological alignment, loose structural overlap, and conflict avoidance. We describe each property and how they are combined next.

Ontological Alignment

The Cyc ontology comes with a number of higher-order relations that relate concepts at an abstract level. For example, functionCorrespondingPredicate indicates that a given function and predicate amount to the same relationship, holding for pairs like temperatureOfObject and TemperatureFn. We use a set of such relationships to estimate expression similarity.

Loose Structural Overlap

We employ a simple, loose measure of structural similarity inspired by the concept of similarity flooding [Melnik *et al*, 2002], due to the distance between the concepts involved.

We first define both sets of logical expressions, *S* and *T*, as hypergraphs. Let *H(T)* be the hypergraph of *T* where hyperedges are expressions and vertices are the entities and variables contained in those expressions. For instance, the expression on Line 10 of Figure 3, would be a hyperedge connecting ?sub, ?thing-melting, and ?self. By considering these sets to be hypergraphs, we can define a notion of neighbors in these hypergraphs.

For a particular expression t_i, we consider the neighboring expressions to be all those expressions with a distance from t_i that is less than a given constant *k*. In our experiments, we use *k* = 3. We define the distance between two expressions t_i and t_j to be the length of the shortest path in *H*(*T*) connecting *any* two entities of t_i and t_j. The loose-structural overlap score of expressions s_i and t_j is 1 if both expressions have neighbors in *H(G)* that are also in *M* and 0 otherwise.

Conflict Avoidance

Our system gives preference to edges that lead to the fewest conflicts overall. Conflicts in the semantic interpretation arise from choices to resolve ambiguities that would lead to logical inconsistencies (i.e. a river bank is not a place to deposit money). The normalized conflict-avoidance score $c(s_i, t_j)$ is the number of edges in *G* that do *not* conflict with edge (s_i, t_j) divided by the cardinality of *G*

Calculating and Using the Score

Let $o(s_i,t_j)$ be the ontological alignment score, $s(s_i,t_j)$ be the loose structural overlap score, and $c(s_i,t_j)$ be the conflict avoidance score. Then, the overal score of a matching *M* between *S* and *T* is given as:

$$\sum_{(s_i,t_j)\in M} \alpha * o(s_i,t_j) + \beta * s(s_i,t_j) + \gamma * c(s_i,t_j)$$

where α, β, and γ are the preferences for each property of the matching. In our experiments, we have $\alpha = 0.4, \beta = 0.4, \gamma = 0.2$ which were values chosen simply because conflict-avoidance was found to be important, but less important than the other two features.

The local-search algorithm obtains a first-pass candidate solution by greedily finding a matching *M* that maximizes for only ontological alignment. From that initial matching *M*, the algorithm begins to iteratively improve its solution by exploring the conflict-free matchings that differ by at most two edges from *M* until it can no longer find a matching with a higher score as determined by the equation above. When a new matching with a higher score is found, the algorithm repeats from that matching, otherwise it returns *M*.

The final matching contains pairings of expressions in *S* with expressions in *T*. Some of those pairs include type constraints. For example, Line 3b of Figure 5 (“water” as Water) might match with Line 3 of Figure 3 (the participant with ChemicalCompoundTypeByChemicalSpecies. From those type constraint pairings, our approach extracts variable bindings (e.g. ?sub with water751940) and instantiates the *T* with those bindings. The result is a subset of the instantiated *T* that includes only participants, participant constraints, and conditions, that is conditioned on the expressions in *S* that were used in *M*.

```
(temperatureOfObject room752597 Warm)      - from matching
(isa water751940 Water)                    - from matching
(isa snow751995 SnowMob)                   - from matching
(isa change2037026 StateChangeEvent)       - from stability analysis
(isa put2038224 PuttingIntoAState)         - from stability analysis
(isa room752597 RoomInAConstruction)       - from stability analysis
```

Figure 7. *Antecedents after matching and stability analysis*

3.1.2 Stability Analysis

Often there are salient features of scenario types that are not indicated by an ontological match. For instance, situations regarding gravity often involve something falling.

Our system creates a SAGE generalization pool for each process type (melting, freezing, gravity, etc.). The generalization pools are populated by the complete sets of semantics for all training instances of that type (e.g. each question about melting is parsed, and its semantics put into a case which is then added to a generalization pool for melting). After the matching is complete, the generalization pool for the given question type is retrieved and is used to assign probabilities to all of the expressions in *S*. The top 3 most probable expressions from *S* that do not conflict with already selected expressions from the matching step are added to our antecedents. Figure 7 shows the antecedents (each of which are predicate calculus interpretations of phrases from our training paragraph) after both the matching and stability analysis and indicates the source of each antecedent.

3.1.3 Steiner Tree Connecting Semantics

At this point, antecedents have been drawn from both the initial matching and generalized stable structures. Referring back to Figure 7, it is clear that the semantics selected thus far are disconnected from one another. We would like to incorporate the context of the question that includes and connects all of those expressions. We pose this as the problem of finding a conflict-free Steiner tree through the hypergraph *H*(*S*) that connects all the entities seen in our selected set of choices from *S* (in Figure 7, those entities would be snow751995, water751940, room752597, etc).

The minimum Steiner tree problem in graphs is the problem of finding the minimum cost tree in a graph *G* that connects a given subset of its vertices. While there are approximation algorithms for the Steiner tree problem e.g. [Agrawal *et al*, 1995], there are no approximation algorithms that take into account negative-disjointness constraints. We use a simple extension to a 2-approximation algorithm for the minimum Steiner tree problem (though we make no optimality guarantees) to ensure it produces a non-conflicting set of semantic choices. While our algorithm is not guaranteed to result in the antecedents becoming fully connected, it appears to work well in practice. It also allows for the straightforward addition of coreference resolution, by extending the set of semantics to include coreference expressions taking two coreferable entities as arguments. The Steiner tree algorithm can connect entities across sentences when necessary by using those expressions.

```
(evidenceForExpression
 (and (isa snow751995 SolidTangibleThing)
      (isa water751940 ChemicalCompoundTypeByChemicalSpecies)
      (substanceOfType snow751995 water751940)
      (relationAllInstance freezingPoint water751940 abduced-temp12)
      (qGreaterThan (TemperatureFn snow751995) abduced-temp12)
 (and (temperatureOfObject room752597 Warm)       - from matching
      (isa water751940 Water)                     - from matching
      (isa snow751995 SnowMob)                       - from matching
      (isa change2037026 StateChangeEvent)     - from stability analysis
      (isa put2038224 PuttingIntoAState)       - from stability analysis
      (isa room752597 RoomInAConstruction)     - from stability analysis
      ((VerbRelFn be) water2037198 snow751995) - from Steiner tree
      (fe_effect put2038224 room752597)          - from Steiner tree
      (fe_Cause put2038224 snow751995)           - from Steiner tree
      ((IBPFn parts) room752597 change2037026) - from Steiner tree
      (objectActedOn change2037026 snow751995))) - from Steiner tree
```

Figure 8. *The final EE (sources of antecedents are to the right)*

3.1.4 Storing Cases

The pairing of choices from *S* and model conditions will be stored in a case library as an evidenceForExpression statement as its own case. Figure 8 shows the final EE produced for the training question in Figure 4, as well as sources for each of the antecedents of the EE. The consequent of the EE are the forms required to instantiate the model fragment (i.e. its participants and constraints).

3.2 Testing

Following training, we have a set of evidenceForExpression statements whose antecedents are question semantics and whose consequents are activation requirements of model fragments (e.g. participants of the correct type and relations between them).

3.2.1 Retrieval and Instantiation

Given a new scenario, it is first interpreted by EA NLU. The complete set of undisambiguated semantics forms a case. MAC/FAC then retrieves the five most similar EEs (i.e. the ones with the most antecedent overlap) from the case library of EEs. The consequents of an EE are inferred by analogical inference and bound with the variables from the scenario interpretation.

EEs are abducible in that only a subset of the antecedents are required to infer the consequent activation conditions. An initial ranking of the EEs is given by the number of abduced antecedents. To prevent over-eager application of EEs to a given scenario, our technique only considers those EEs with at least as many antecedents satisified as abduced.

3.2.2 Composing EEs

The ordered EEs are input into a slight variant of the query composition algorithm of Crouse *et al* [2018]. The modification excludes conflict counts from being considered in the score of an EE, which turns the composition algorithm into a greedier coverage-focused algorithm (i.e. an algorithm that looks for EEs covering as many semantic choices as possible, without regard to how many semantic choices the selected EEs would rule out through choice-set constraints). We briefly recap the algorithm here.

The composition algorithm takes a set of instantiated EEs and a set of semantics *S* to be covered. The antecedents of each EE are elements of *S*, where *S* is the set of semantic choices for the current scenario. The algorithm iteratively selects the EE whose antecedents cover as much of *S* is possible, removing semantic choices from *S* that conflict with selected EEs as it goes. When *S* is empty (because the elements of *S* were either covered by some selected EE or conflicted with the antecedents of some selected EE) the algorithm returns the consequents of every selected EE. The appeal of the algorithm is that it treats the EE selection process as a coverage problem, producing the smallest set of consequents that reflects as much of the semantics as is possible.

3.2.3 Model Formulation and Question-Answering

The output of the composition algorithm is a set of activation conditions that can be used to instantiate model fragments relevant to the scenario at hand. Our technique uses this model formulation algorithm to instantiate all applicable model fragments and collects the resultant facts about the scenario into a set of model facts *F*.

Our method is evaluated on two types of scenario questions: standard questions (e.g. those that end in a question mark like "What process occurred?") and fill-in-the-blank questions (e.g. "When ice melts it ____"). To handle standard questions, we first filter out all expressions from *F* that do not have a positive alignment score (like in training) with our question semantics. Then, for each answer option, the semantics for the answer are matched against *F* using the same matching procedure described in training. For fill-in-the-blank questions the process is largely the same. For each answer option, the semantics of the question *and* answer are matched against *F* using the matching procedure from training, and the highest scoring answer of the question-answer pairs is output as the answer to the question. If no models can be instantiated, then no answer will be chosen.

4 Experimental Evaluation and Discussion

We evaluate our approach on a set of 45 questions from 4th and 5th grade elementary science tests. This set of questions was extracted by a script that searched across a large set of science test questions (collected by the Allen Institute for Artificial Intelligence). The questions the script returned had keywords associated with the model fragments outlined in Crouse and Forbus' [2016] science test analysis. Furthermore, the questions were restricted to those involving reasoning about a scenario, not questions involving definitions or taxonomies. Future work will involve determining all of the phenomena in the elementary science tests that can be representable by QR models.

Question Type	Correct / Total	Percent
All	26 / 45	58%
Model Formulation	19 / 26	73%
Model Reasoning	7 / 19	37%

Table 1. *Average Performance*

The questions our system was designed to handle were those only requiring model formulation. Those involving more complex QR techniques like qualitative simulation or differential qualitative analysis were out of the scope of this work. We categorized those questions requiring only model formulation as *model formulation* questions and those requiring additional reasoning on top of relevant instantiated model fragments generated by model formulation (i.e. all others) as *model reasoning* questions.

We evaluated using 5-fold cross validation (i.e. 36 questions for training, 9 questions for testing per fold). Table 1 shows our overall average performance on all questions, model formulation questions, and model reasoning questions. Random guessing on this dataset would lead to 25% correct.

The performance gap between model formulation and model reasoning questions is quite substantial. This comes as no surprise, given that our approach is not yet learning how to answer more reasoning-intensive questions during training. This gives one immediate avenue for future work, which is to extend our approach to learn the reasoning needed to answer the more advanced questions through only question-answer pairs.

4 out of 7 errors for model formulation questions were due to phrasings outside of the handling of our approach. For example, the question, "Which type of force requires contact between two objects for one to …" requires one to know that "two objects" implies there are two distinct objects in the scenario. Our approach only identifies one object from that scenario, and thus cannot successfully instantiate a model fragment for friction. The remaining 3 errors were due to inadequate training data, where our approach hadn't seen a scenario similar enough to formulate the correct model to answer the question.

The model reasoning questions gave our approach much more difficulty. Apart from requiring more sophisticated reasoning techniques, the language of the questions tended to be more complex. Accordingly, for 4 out of 19 questions our approach found the correct model fragment to use, but instantiated it incorrectly. For 6 out of 19 questions the wrong model fragment was selected, while in another 6 out of 19 questions the correct model fragment was instantiated. The unfortunate last source of issues were processes that were never seen during training. Our corpus only had two questions revolving around fluid displacement, and both of those questions were found in the same fold.

5 Related Work

Barbella & Forbus [2011] introduced *analogical dialogue acts* (ADAs), which formalize the roles played by individual utterances in instructional analogies. Their approach used the ADAs recognized from the semantic parse of an instructional text to build structured cases that were then compared with SME. Their system used inferences from these analogies to interpret and answer questions. Our approach also uses analogical inferences to construct an interpretation of text (scenario model), however our system goes a step further in that EEs are learned from natural language while ADAs were recognized with manually constructed rules. They also used dynamic case construction [Mostek *et al*, 2000] to automatically extended their cases with pertinent background facts from the knowledge base, which may be a useful technique to incorporate into our system to validate activation conditions which may be available as stored knowledge.

Barbella & Forbus [2015] further present a method for constructing coherent cases from text which is similar to our goal of extracting relavent semantics. One way our approaches differ is that our relevence condition is overlap with a pre-existing model while theirs finds facts related to a seed from the text. Their approach is complementary to ours and could be used to segment a corpus into cases from which our approach could build interpretations.

Chang [2016] combined natural language understanding, spatial reasoning, and analogical reasoning to interpret instructional analogies. These analogies could be used to learn qualitative knowledge. Of particular relevance to our work was the use of visual representations to disambiguate natural language. Their work used the CogSketch sketch understanding system [Forbus *et al*, 2011] to represent sketches with the Cyc ontology. EA NLU semantic choice-sets were disambiguated by selecting those choices that were most related to the outputs of the sketch understanding system. This could be incorporated naturally into our work as part of the ontological features our approach uses during the matching step.

Khashabi *et al* [2017] introduced the notion of *essential question terms*, which were terms absolutely critical to the understanding of a particular question. They showed that without those terms, human performance on science test questions dropped significantly. This is related to our approach which learns the essential components of a scenario needed to infer the activation conditions of a particular model fragment (EEs).

Khot *et al* [2017] introduced a method for answering complex, compositional science test questions from OpenIE extracted knowledge bases. They posed the problem of multiple-choice question-answering as a search for an optimal subgraph connecting a question and answer through the knowledge base. The types of questions this system was designed for were compositional factoid questions, which likely makes their system complementary to ours.

Fan and Porter [2004] introduced Loose-speak, an interpreter intended to fix misalignments commonly seen between the queries of novice users of a knowledge base and the knowledge base being queried. It was equipped with a variety of features for determining when an expression in the knowledge base was likely the intended expression of the user, some of which are similar to the ontological alignment features of our work.

6 Conclusions and Future Work

We have described an implemented system that adapts a domain-general semantic parser to build qualitative scenario models. Our system uses these models to answer elementary science test questions. Our approach builds on prior work by Crouse *et al* [2018] and uses their EE formalism for cases that are retrieved and applied by analogy to adapt the parser's outputs. The primary contribution of this work is a novel technique which associates the relevant semantics of a scenario to qualitative model fragments and applies these associations by analogy to construct a scenario model. This approach goes significantly beyond prior work in its ability to build models for multi-sentence scenario questions and in providing a richer framework for judging ontological similarity.

There are three clear directions for future work. First, our approach only utilized a small number of qualitative models and thus it was limited in scope. Future work will use the mapping between the core elements of QP theory and FrameNet produced in [McFate & Forbus, 2016a] to extract qualitative models *a la* [McFate & Forbus, 2016b] and broaden the range of model fragments our system can utilize. Second, we will extend the reasoning capabilities of our system to handle a larger range of questions. Finally, we do not currently generate natural language responses, nor natural language explanations for the answers given. Generating both would set the stage for teaching and correcting the system via interactive dialogue.

Acknowledgements

This research was supported by the Intelligent and Autonomous Systems Program of the Office of Naval Research

References

[Agrawal *et al*, 1995] Agrawal, A., Klein, P., & Ravi, R. (1995). When trees collide: An approximation algorithm for the generalized Steiner problem on networks. *SIAM Journal on Computing*, *24*(3), 440-456.

[Barbella & Forbus, 2011] Barbella, D. and Forbus, K. (2011). Analogical Dialogue Acts: Supporting Learning by Reading Analogies in Instructional Texts. *In Proc. AAAI-11*, San Francisco, CA.

[Barbella & Forbus, 2015] Barbella, D. and Forbus, K. (2015). Exploiting Connectivity for Case Construction in Learning by Reading. *In Proc. of the Third Annual Conference on Advances in Cognitive Systems.*

[Chang, 2016] Chang, M. (2016). Capturing Qualitative Science Knowledge with Multimodal Instructional Analogies (Doctoral dissertation, Northwestern University).

[Clark *et al*, 2016] Clark, P., Etzioni, O., Khot, T., Sabharwal, A., Tafjord, O., Turney, P. D., & Khashabi, D. (2016, February). Combining Retrieval, Statistics, and Inference to Answer Elementary Science Questions. *In Proc. AAAI-16* (pp. 2580-2586).

[Crouse & Forbus, 2016] Crouse, M., & Forbus, K. D. (2016). Elementary School Science as a Cognitive System Domain: How Much Qualitative Reasoning is Required?. *Advances in Cognitive Systems*, *19*.

[Crouse *et al*, 2018] Crouse, M., McFate, C., & Forbus, K. (2018). Learning from Unannotated QA Pairs to Analogically Disambiguate and Answer Questions. In *Proc. AAAI-18*

[Darmann *et al*, 2011] Darmann, A., Pferschy, U., Schauer, J., & Woeginger, G. J. (2011). Paths, trees and matchings under disjunctive constraints. *Discrete Applied Mathematics*, *159*(16), 1726-1735.

[Fan & Porter, 2014] Fan, J., & Porter, B. (2004, July). Interpreting loosely encoded questions. In *Proc AAAI-14* (pp. 399-405).

[Fillmore *et al*, 2001] Fillmore, C. J., Wooters, C., & Baker, C. F. (2001). Building a large lexical databank which provides deep semantics. In *PACLIC-15* (pp. 3-26).

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

[Forbus *et al*, 1995] Forbus, K. D., Gentner, D., & Law, K. (1995). MAC/FAC: A model of similarity-based retrieval. *Cognitive science*, *19*(2), 141-205.

[Forbus *et al*, 2011] Forbus, K., Usher, J., Lovett, A., Lockwood, K., & Wetzel, J. (2011). CogSketch: Sketch understanding for cognitive science research and for education. *Topics in Cognitive Science*, *3*(4), 648-666.

[Forbus *et al*, 2017] Forbus, K. D., Ferguson, R. W., Lovett, A., & Gentner, D. (2017). Extending SME to Handle Large-Scale Cognitive Modeling. *Cognitive Science*, *41*(5), 1152-1201.

[Gentner, 1983] Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive science*, *7*(2), 155-170.

[Khashabi *et al*, 2016] Khashabi, D., Khot, T., Sabharwal, A., Clark, P., Etzioni, O., & Roth, D. (2016, July). Question answering via integer programming over semi-structured knowledge. *In Proc. IJCAI*-16. AAAI Press.

[Khashabi *et al*, 2017] Khashabi, D., Khot, T., Sabharwal, A., & Roth, D. (2017). Learning What is Essential in Questions. In *Proc. CoNLL 2017)* (pp. 80-89).

[Khot *et al*, 2017] Khot, T., Sabharwal, A., & Clark, P. (2017). Answering Complex Questions Using Open Information Extraction. In *Proc. ACL-17*.

[Matuszek *et al*, 2006] Matuszek, C., Cabral, J., Witbrock, M. J., & DeOliveira, J. (2006, March). An Introduction to the Syntax and Content of Cyc. In *AAAI Spring Symposium* (pp. 44-49).

[Mostek *et al*, 2000] Mostek, T., Forbus, K, and Meverden, C. (2000). Dynamic case creation and expansion for analogical reasoning. *In Proc. of AAAI-2000*. Austin, TX.

[McFate & Forbus, 2011] McFate, C. J., & Forbus, K. D. (2011, June). NULEX: an open-license broad coverage lexicon. In *Proc. NACL-HLT-11*

[McFate & Forbus, 2016a] McFate, C., & Forbus, K. (2016). *An Analysis of Frame Semantics of Continuous Processes*. *In Proc. CogSci-16*

[McFate & Forbus, 2016b] McFate, C., & Forbus, K. (2016). Scaling up Linguistic Processing of Qualitative Processes. *In Proc. Advances in Cognitive Systems 2016.*

[McLure *et al*, 2015] McLure, M. D., Friedman, S. E., & Forbus, K. D. (2015, January). Extending Analogical Generalization with Near-Misses. In *Proc. AAAI-15* (pp. 565-571).

[Melnik *et al*, 2002] Melnik, S., Garcia-Molina, H., & Rahm, E. (2002). Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *Data Engineering, 2002. Proceedings. 18th International Conference on* (pp. 117-128). IEEE.

[Schoenick *et al*, 2017] Schoenick, C., Clark, P., Tafjord, O., Turney, P., & Etzioni, O. (2017). Moving beyond the turing test with the allen ai science challenge. *Communications of the ACM*, *60*(9), 60-64.

[Tomai & Forbus, 2009] Tomai, E., & Forbus, K. D. (2009, March). EA NLU: Practical Language Understanding for Cognitive Modeling. In *FLAIRS Conference 2009*.