Capturing QP-relevant Information from Natural Language Text

Sven E. Kuehne (skuehne@northwestern.edu)
Kenneth D. Forbus (forbus@northwestern.edu)
Qualitative Reasoning Group, Northwestern University
1890 Maple Avenue, Ste. 300
Evanston, IL 60201, USA

Abstract

People can learn about the physical world from textbooks and develop an understanding of physical phenomena from simple descriptions. As part of our ongoing investigation of the extraction and representation of knowledge about physical processes found in natural language text, we describe a natural language system that captures information about instances of physical processes from paragraph-sized descriptions through a deep semantic interpretation process as a set of interconnected frame structures.

Introduction

When people read descriptions of physical phenomena in textbooks they usually have certain expectations about the information and mentally construct appropriate models of the described phenomena. This construction is an idiosyncratic process, because the readers need to interpret the author's description, building their own model of it. Readers have to use their background knowledge to eliminate potentially ambiguous interpretations and to fill gaps left by the natural language description. In other words, reading about physical processes involves interpreting the text by constructing a model. In the best case, it is an exact reconstruction of the author's intended model of the process.

QP theory (Forbus, 1984) concerns the structure of a class of physical theories, and has been successfully used in a variety of reasoning systems (Forbus, 1996). The hypothesis is that many mental models of physical phenomena can be expressed in this formalism. QP theory has been used to develop a wide range of models of phenomena, including economics, ecology and medicine in addition to physical models. This makes it an excellent candidate for a component in a larger system of natural language semantics.

The fact that humans can learn about the physical world from textbooks and other sources leads to a number of interesting questions about the connections between our conceptual understanding of the physical world and how it is reflected in natural language. If students can learn from simple descriptions of physical phenomena, can the knowledge included in these texts be extracted to automatically construct models of the underlying physical processes?

Understanding descriptions of physical phenomena starts with the identification of continuous parameters that are involved in the physical processes. Descriptions of physical phenomena typically contain abundant references to physical quantities. The extraction of information about continuous parameters is therefore an essential step in building models of physical processes (Kuehne, 2003).

We have previously shown that natural language descriptions of physical phenomena can contain abundant QP-relevant information (Kuehne & Forbus, 2002). If one looks carefully at the descriptions of physical processes, given either as a concrete example or as generalized knowledge, one can identify parts of the natural language description that correspond to certain elements of QP Theory. In the same context, we have also proposed that QP theory can provide a knowledge representation language for aspects of natural language semantics concerned with continuous parameters and continuous causation. We have outlined a representational scheme that recasts the theoretical framework of QP theory in terms of frame semantics. QP frames use a representational scheme that is compatible with the notions of frames and frame elements in FrameNet (Baker, Fillmore, & Lowe, 1998; Fillmore, Wooters, & Baker, 2001). They form an intermediate representational layer between the natural language input and the representations that can be used in qualitative reasoning, e.g. model fragments in CML (Falkenhainer et al., 1994).

In this paper we describe a natural language system that captures QP-relevant information from descriptions of instances of physical processes. It uses a deep semantic interpretation process and represents the contents as a set of interconnected QP frame structures. The system combines the results of our previous work, it is fully implemented, and has been tested on a dozen paragraph-sized natural language descriptions (Kuehne, 2004). We begin with an overview of the system and its individual components. A detailed example illustrates the interpretation process, followed by a comparison of the information captured in terms of QP frame structures with a manually constructed process model. Finally, we discuss a number of problems that we encountered in using the system to capture QP-relevant information from natural language text and plans for future work.

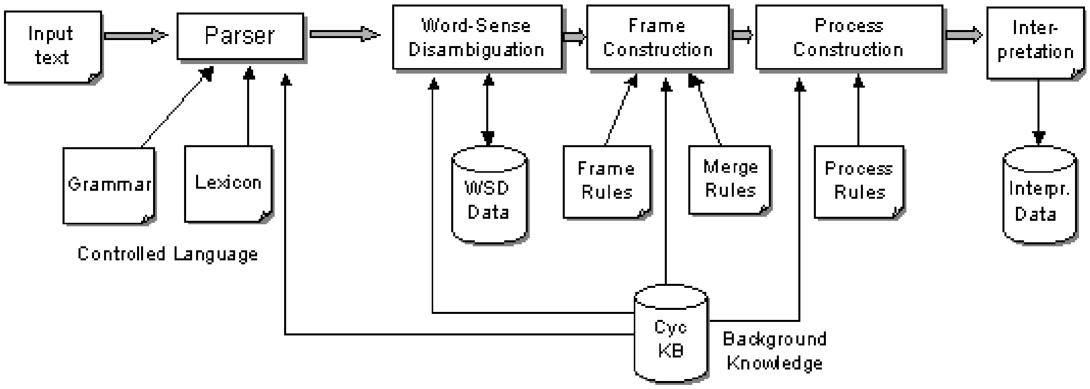


Figure 1: Architecture overview

Overview

Almost every natural language processing system that produces deep semantic representations from textual descriptions uses a syntactic parser and an interpretation process for analyzing the results of the syntactic analysis (cf. Barker, Delisle, & Szpakowicz, 1998; Mahesh & Nirenburg, 1995; Nyberg & Mitamura, 1992). We followed this modular approach and designed a system that builds models of physical processes from natural language descriptions in two distinct steps. First, the input is subjected to a syntactic analysis by the parser, which generates a list of parse trees that correspond to all possible syntactic interpretations based on the grammar it uses. The results of the parsing step are then used in a semantic interpretation process to produce particular domain-specific representations of the descriptions. Figure 1 shows an overview of the architecture of our system.

The *input documents* to our system consist of multi-sentence descriptions of instances of physical processes written in a controlled subset of standard English. The fact that unrestricted natural language is full of ambiguity, even when the domain itself imposes some constraints, presents a challenge to any NL system that tries to extract information from text. Ambiguity can arise from word meanings, e.g. the polysemy of individual words or the interpretation of word compounds, and from grammatical constructs, e.g. multiple interpretations of a sentence based on different prepositional phrase attachment. Sentences like ‘Fruit flies like bananas’ or ‘I saw the man on the hill with a telescope’ are classic examples that illustrate the ambiguity of natural language.

The use of a controlled language can reduce ambiguity by restricting the grammar and the lexicon. Controlled languages have a long history that predates the fields of computational linguistics and natural language understanding (Ogden, 1933) and have found applications in technical domains such as the preparation of technical documentation (Almquist & Sagvall Hein, 1996; Wojcik, Holmback, & Hoard, 1998), logic representations of operating procedures (Fuchs & Schwitter, 1996), and knowledge-based machine translation (Mitamura & Nyberg, 1995).

To facilitate the interpretation of the input material, we have designed QRG Controlled English (QRG-CE) as a *controlled language* for describing physical phenomena in a readable, yet less ambiguous subset of English. While QRG-CE does not impose any restrictions on the lexicon and allows multiple word meanings, it uses grammatical restrictions to reduce syntactic ambiguity. QRG-CE includes the syntactic realizations of QP theory concepts in natural language (Kuehne & Forbus, 2002) as grammatical rules.[1]

The *parser* is a modified version of the publicly available parser described in (Allen, 1995). Its bottom-up parsing algorithm constructs an interpretation of a sentence in a compositional manner, starting from terminal nodes. Using a best-first parsing technique, it attempts to maximize the length of phrases and sentence structures it can handle. The parser supports partial parsing, i.e. even when no syntactic analysis of the entire input sequence is possible, the parser generates interpretation for phrases and individual words. Information such as the semantic data for a node is ‘moved up’ to the phrase head when new phrases are constructed from constituent nodes by using feature percolation. Depending on the number of possible syntactic analyses, i.e. distinct parse trees, the resulting interpretation can consist of one or more sets of expressions. The parser itself produces a general semantic interpretation based on the grammar rules for the controlled language and general semantic information retrieved from the background knowledge base. The *background knowledge base* consists of a subset of the Cyc Knowledge Base (Lenat & Guha, 1989). The resulting interpretation does not contain any QP frame structures

[1] As of May 2004, 19 out of 107 rules in our controlled grammar contain support for QP-relevant content. This information is used in the interpretation process that follows the syntactic analysis of the input descriptions.

yet, but it includes supporting information about QP-specific patterns. Moreover, the general semantic interpretation data can contain a certain degree of ambiguous information, which requires a semantic interpretation process to eliminate the remaining ambiguity and produce the best possible interpretation of a sentence. This includes the disambiguation of multiple word meanings and enforcing the preference for domain-specific constructs.

Ambiguous conceptual information included in the general semantic interpretation data is resolved by a *word-sense disambiguation* module. For example, the semantic information attached to the noun 'bar' can include the concepts corresponding to 'drinking establishment' and 'unit of pressure'. Based on evidence such as contextual information and domain-specific constraints, the word-sense disambiguation process will prefer one concept over another. Using third-party resources such as the Cyc KB contents, we have to deal with inconsistencies such as missing entries, non-aligned argument structures and erroneous part of speech information. The system employs a word-sense disambiguation module that uses an evidence-based approach to collect and weigh different types of evidence supporting individual word senses.

Support for a particular word sense falls into four major categories: tests for task-specific evidence, tests for contextual restrictions, tests based on preferences in the knowledge base, and user preferences. Task-specific evidence is based on the relevance of a concept for the domain in which the system is operating. For the interpretation of physical phenomena, the system is looking for information that relates a concept to a known quantity type, a physical process, or domain-specific terminology. For example, the knowledge base contains the concepts `Hot`, `Hot-Spicy` and `GoodLooking` for the adjective 'hot'. The concept `Hot` is preferred over its competitors because it refers to a quantity type. Selectional restrictions are used as contextual evidence if a concept fits the slot of an expression in which it is used. If the predicate `emptiesInto` requires one of its arguments to be a 'stream', the concept `River` is considered more relevant than `DataStream` because it matches the selectional restrictions for the argument slot. Since the knowledge base can contain inconsistencies, slot restrictions cannot be used as a hard constraint. Preferences for certain word senses are based solely on the organization of the KB contents, e.g. the preference for specializations over their superclasses and are treated as weak evidence with lower weights. Finally, user preferences gathered from previous manual word sense disambiguation can be counted as evidence.[2]

A *QP-specific semantic interpretation* step then constructs QP frame structures from the disambiguated general semantic information via sets of forward-chaining rules (Forbus & de Kleer, 1993). For example, the following rule recognizes a possessive relation between two objects, as in the noun phrase *'the temperature of the brick'.*

```
(rule ((:true (possessiveRelation ?owner ?thing)
              :var ?pr)
       (:true (isa ?thing ?qtype)
              :var ?isqtype
              :test (quantity-type-p ?qtype)))
   (let ((?qframe (make-frameid 'QuantityFrame)))
      (rassert!
       (:implies (:and ?pr ?isqtype)
         (:and (isa ?qframe QuantityFrame)
               (entity ?qframe ?owner)
               (quantityType ?qframe ?qtype))))))
```

The rule instantiates the appropriate expressions for a Quantity frame, if its conditions are met. Note that the rule only generates slot expressions for the entity and the quantity type. This is the minimal information required for the instantiation of a quantity. Other rules can add further slot information, e.g. about the sign of derivative to the Quantity frame. This technique allows the interpreter to build frame information in a incremental fashion, even across multiple sentences.

The system can process paragraph-sized descriptions of process instances by merging frame information. Similar to the construction of QP frames for individual sentences, the interpreter uses forward-chaining rules to detect mergeable frames. As a final step, the semantic interpretation process identifies all QP frames belonging to a particular physical process and creates the appropriate PhysicalProcess frame structures.

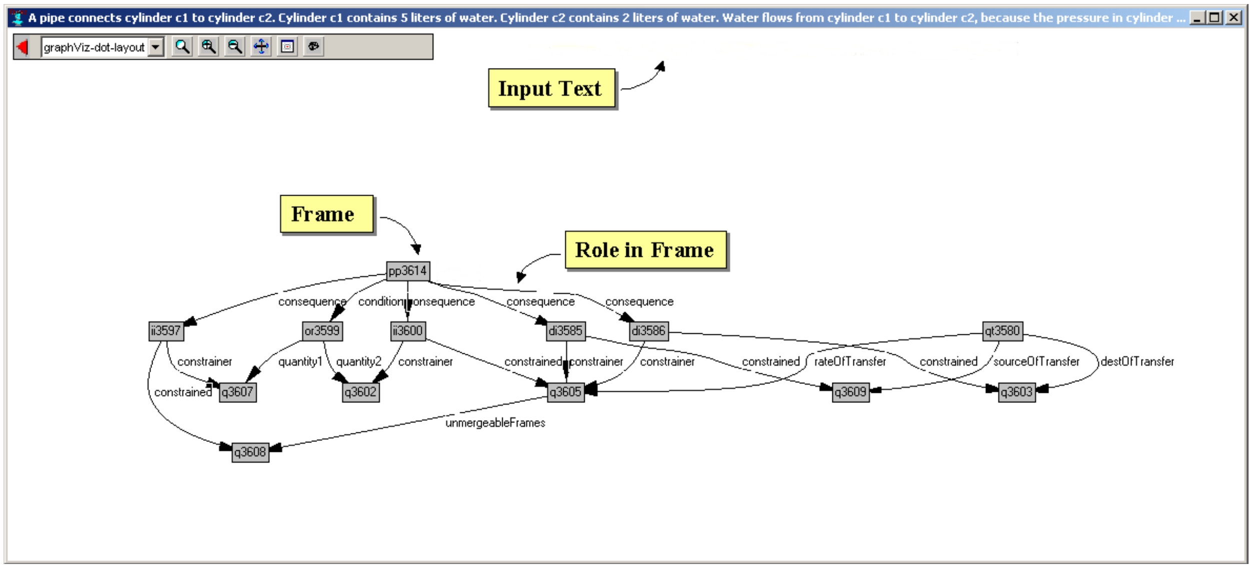
The following section uses a multi-sentence description of a classic QP scenario to illustrate how information about the underlying physical process can be captured by the semantic interpreter in terms of QP frames. The process frame information is then compared against a hand-coded model for the same description.

Example: Fluid flow between two containers

Here is how a classic QR example, a fluid flow between two containers, can be described in our controlled language:

(1) A pipe connects cylinder c1 to cylinder c2.
(2) Cylinder c1 contains 5 liters of water.
(3) Cylinder c2 contains 2 liters of water.
(4) Water flows from cylinder c1 to cylinder c2, because the pressure in cylinder c1 is greater than the pressure in cylinder c2.
(5) The higher the pressure in cylinder c1, the higher the flowrate of the water.

[2] The system can be run in a manual training mode, in which the user picks the appropriate word sense. The choices are recorded and can be used in the automatic disambiguation process. However, for the example presented in this paper and in (Kuehne, 2004) the system did not use user-specific training data.



A pipe connects cylinder c1 to cylinder c2. Cylinder c1 contains 5 liters of water. Cylinder c2 contains 2 liters of water. Water flows from cylinder c1 to cylinder c2, because the pressure in cylinder ...

Figure 2: QP frame structures for the two-container example

(6) When the pressure in cylinder c2 increases, the flowrate of the water decreases.

The first three sentences establish the scenario used for the fluid flow between two containers. The two cylinders are named by using the labels ‘c1’ and ‘c2’ instead of automatically generated, generic discourse names for each cylinder instance.

Sentence 4 describes the actual flow event between the containers. It also explicitly names the level difference as the cause for the flow. Also note that the two level quantities are compared directly in this example. Sentences 5 and 6 use typical syntactic patterns for indirect influences that describe qualitative proportionalities.

Figure 2 shows the frame structures generated by the semantic interpreter for this example. The nodes of the graph represent the QP frames, while the edges mark the frame elements, i.e. the roles played by a constituent frame in a parent frame.[3] The rest of this section examines the full set of QP frames produced by the semantic interpreter for this example.

Four quantity frames are generated for the amounts of water and the pressure in the cylinders.

```
Frame q3609 (QuantityFrame)
  Entity: c1
  QType:  (AmountFn (LiquidFn Water))
  Value:  5
  Unit:   Liter

Frame q3603 (QuantityFrame)
  Entity: c2
  QType:  (AmountFn (LiquidFn Water))
  Value:  2
  Unit:   Liter


Frame q3607 (QuantityFrame)
  Entity: c1
  QType:  Pressure
  Sign:   Positive

Frame q3602 (QuantityFrame)
  Entity: c2
  QType:  Pressure
  Sign:   Positive
```

Two more quantity frames are generated for the flowrate. Note that the system creates two quantity frames for a flowrate. Although the flowrate mentioned in sentence 6 should be identical with the previously mentioned flowrate (indicated by the fact that they both refer to the same entity, `flow3606`), the system does not merge these frames because their signs of derivative are different. The two rate frames are linked by an `unmergeableFrames` relation.

```
Frame q3608 (QuantityFrame)
  Entity: flow3606
  QType:  Rate
  Sign:   Positive

Frame q3605 (QuantityFrame)
  Entity: flow3606
  QType:  Rate
  Sign:   Negative
```

A single OrdinalRelation frame is created for the different levels in the cylinders, because the comparison in (4) is directional. The sentence explicitly states that the level in C1 is greater than the level in C2. The interpreter can also construct OrdinalRelation frames for implicit comparisons. For example, for the flow of heat from a hot object *A* to a

[3] A detailed overview of the QP frames and their frame elements can be found in (Kuehne & Forbus, 2002).

```
(defModelFragment waterflow                              (isa flow3606 Translation-Flow)
  :subclass (flow)
  :participants ((src :type contained-stuff)             (isa c1 Container) [QuantityFrame 3609]
                 (dst :type contained-stuff)             (isa c2 Container) [QuantityFrame 3603]
                 (con :type path)
  :conditions ((connects con src dst)                    (> (pressure c1) (pressure c2))
               (> (pressure src) (pressure dst))
  :quantities ((flowrate :dimension rate-dimension))     [QuantityFrames q3608 and q3605]
  :consequences ((Qprop+ (flowrate :self)
                         (pressure src))                 (qprop (flowrate flow3606) (pressure c1))
                 (Qprop- (flowrate :self)                (qprop- (flowrate flow3606) (pressure c2))
                         (pressure dst))
                 (I- (water src) (flowrate :self))       (I- (water c1)) (flowrate flow3606))
                 (I+ (water dst) (flowrate :self))))     (I+ (water c2)) (flowrate flow3606))
```

Figure 3: Manually constructed model for comparison

cool object *B* the interpreter generates two OrdinalRelation frames for `(> (temp A) (temp B))` and `(< (temp B) (temp A))`.

```
Frame or3599 (OrdinalRelationFrame)
  Quantity1: q3607
  Quantity2: q3602
  Relation:  greaterThan
```

The transfer of water between the two cylinders is captured by a QuantityTransfer frame, which identifies the amount of water in cylinder C1 as the source and the amount of water in C2 as the destination quantities of the flow. Since no explicit rate is mentioned yet, the semantic interpreter instantiates a default Quantity frame for the rate. The information from the QuantityTransfer frame is then used to generate the appropriate DirectInfluence frames for the flow.

```
Frame qt3580 (QuantityTransferFrame)
  Source: q3609
  Dest:   q3603
  Rate:   q3605

Frame di3586 (DirectInfluenceFrame)
  Constrained: q3603
  Constrainer: q3605
  Sign:        Positive

Frame di3585 (DirectInfluenceFrame)
  Constrained: q3609
  Constrainer: q3605
  Sign:        Negative
```

The information about the qualitative proportionalities described in (5) and (6) leads to the instantiation of two IndirectInfluence frames, capturing the influence of the pressure in the cylinders on the flowrate of the water. The interpretation of (5) includes a Quantity frame for the flowrate of water, which is merged with the default rate frame instantiated by the previous sentence. The flowrate is mentioned again in (6), but since it has a different sign of derivative, the quantity information is not merged with the previous rate frame.

```
Frame ii3597 (IndirectInfluenceFrame)
  Constrained: q3608
  Constrainer: q3607
  Sign:        Positive

Frame ii3600 (IndirectInfluenceFrame)
  Constrained: q3605
  Constrainer: q3602
  Sign:        Negative
```

The resulting PhysicalProcess frame includes the frame for direct and indirect influences as consequences and the OrdinalRelation frame as a condition.

```
Frame pp3614 (PhysicalProcessFrame)
  Type:
    Translation-Flow
    PhysicalProcess
  Participants:
    c2
    c1
  Conditions:
    or3599
  Consequences:
    di3586
    ii3597
    ii3600
    (toLocation flow3606 c2)
    (fromLocation flow3606 c1)
    di3585
  Status:
    Active
```

A comparison of the contents of process frames with the information contained in hand-coded models is useful for the evaluation of the semantic interpretation results produced by our system. Figure 3 shows a model fragment for a water flow process and the data for instantiation of the two-container example.[4]

The interpretation data generated by our system captures most of the information contained in the model fragment and the scenario definition. The PhysicalProcess frame for

[4] Corresponding pieces of information in the model fragment and the interpretation data are juxtaposed and color-coded.

the water flow includes both of the cylinders as participants as well as the direct and indirect influences and the ordinal relationship between the different levels as a condition. The third participant in the model fragment, the pipe, is missing in the PhysicalProcess frame. The pipe is only indirectly involved in the actual flow process as a connection between the two cylinders. The interpretation data contains expressions for a connection event that captures this relationship. However, the description does not explicitly mention the connection as a condition for the flow process, and the expressions are therefore not associated with the PhysicalProcess frame.

The two flowrate frames as an internal quantity of the model fragment is also present in the information extracted from the process description, captured by two rate quantities associated with the flow event. Although the two rate frames are marked as unmergeable in the interpretation data for the described instance of a physical process, the frames should be merged in the generation of abstract model fragment information. We will address this point in a future extension to our system.

For consequences of the process, the interpreter generates the appropriate QP frames for the direct influences and includes them in the PhysicalProcess frame. Note that information about the two indirect influences in sentences 5 and 6 is symmetric. This information has to be explicitly stated, since the interpreter does not conjecture additional frame data from partial or incomplete descriptions.

Incomplete information, such as implicit world knowledge or assumptions, is a common phenomenon in descriptions of physical processes. For example, human readers easily assume that an unblocked connection between the two cylinders is necessary for the flow. Authors also try to avoid repetitions and leave out parts that are similar or symmetrical to others, as in (5) and (6). The readers are required (and usually able) to 'fill in' these parts based on their background knowledge. While this technique might work for a human reader, it poses a much greater challenge for a computer system. Nevertheless, by accumulating a number of different descriptions of the same processes, the information missing in individual descriptions might be added and a more complete general model could be constructed from individual descriptions through a generalization process (Kuehne, Forbus, Gentner & Quinn, 2000).[5]

Interpretation issues

As the example illustrates, missing information in the description and incomplete background knowledge are the two main causes for incomplete models. Writers of textbooks and popular science literature often assume some familiarity with basic world knowledge. The author can therefore leave out some parts of the descriptions and expect the reader to 'fill in the blanks.' A similar assumption cannot yet be made when the text of a single description is processed by an automated system.

Some grammatical limitations of the controlled language make the rewriting process slightly complicated. Among them are the missing support for coordinated conjunctions, as in *'the water and the oil are flowing though the pipe'*, compound nouns ('water vapor'), passive constructs, such as '*the ball is placed in the box*', and the support for different verb tenses, temporal ordering ('after'), and measures ('daily', 'percent'). This kind of information is currently lost in the rewritten text, but overcoming these limitations are planned future extensions to the system.

Difficulties are also caused by the fact that proper nouns and terminology need to be defined in the lexicon before the parse is attempted. If a proper noun is not defined, it will either be treated as a label, if it appears together with a common noun in a noun phrase, or as an unknown word, which will most likely prohibit the construction of a complete parse tree for the current sentence. The first outcome can be used as an interesting workaround for the requirement of prior definitions. The proper noun can be used as variable in conjunction with a common noun, e.g. 'the man Joe' instead of just 'Joe'. In this case, the interpretation process will treat 'Joe' as an instance of the concept man and associate all the relevant semantic information from the knowledge base with it, i.e. `(isa Joe AdultMalePerson)`. Special cases of undefined but frequently encountered words are compounds such as 'relative humidity' or 'heat engine'. These terms are defined in the lexicon as hyphenated entries, such as 'heat-engine' or 'relative-humidity'. It should be noted that interpreting compound noun phrases is notoriously difficult (cf. Wisniewski & Love, 1998) and this would be a problem that is quite likely to require learning conventional interpretations.

To find out how extensive the problem of undefined proper nouns and the lack of domain-specific vocabulary is, we have analyzed a representative part of the corpus material for words not covered by our lexicon. The *Sun Up to Sun Down* part (Buckley, 1979) of our corpus contained 93 missing words (out of a total of 3,319 words, or 2.8%) that were not part of the COMLEX 3.1 lexicon data (Macleod, Grishman, & Meyers, 1998) used by our parser. More than half of these words (53, or 56.38%) were hyphenated compounds, such as 'house-heating' or 'water-filled'. The remaining missing entries were mostly place names and adjectives such 'Australia' or 'Irish', and technical terms such as 'absorber' or 'biomass'.

While missing lexical entries manifest themselves primarily in incomplete parses, a major limitation that often prevents a successful semantic analysis of the input text is the link between the lexicon used by the parser and the list of lexical items defined in the Cyc KB. While the

[5] This assumes that there is complementary information between different descriptions, as well as sufficient overlap to make them similar to each other.

parser lexicon contains 86,297 expanded entries based on 39,533 unique entries in the COMLEX data, the Cyc lexicon defines merely 16,552 instances of the collection `EnglishWord`.[6] Even if the same word is defined in both lexicons, orthographic differences can still prevent a successful mapping.

Another common source of problems are unconnected lexical entries and undefined concepts in the knowledge base. Lexicon entries are unconnected if some lexical information is missing that would be required for finding the appropriate concept, such as missing part of speech data or denotational information. We have encountered several instances in which a lexicon entry and appropriate concept definitions existed in the knowledge base but were not linked. In other cases, part of speech information was omitted for a word sense, preventing a successful lookup of semantic information for a word.

For some lexicon entries the knowledge base does not contain any defined concept at all, i.e. no denotational information is associated with a particular word in the lexicon. The result is the same as for unconnected lexical entries, i.e. no concept or semantic information can be retrieved from the knowledge base.

Underdefined concepts are less problematic. Even if a concept can be retrieved for a particular lexical entry, we have often found no attached semantic information. For nouns, adjectives, and adverbs this usually is not a real problem, as long as the concept is tied correctly into the ontology. However, for verbs and prepositions the additional semantic information is important, since it ties different pieces of information within a sentence together during the construction of phrase nodes when keywords in the semantic data are replaced by discourse variables. For nearly half of the 2062 words[7] occurring in *Sun Up to Sun Down* our knowledge base did not contain any semantic information (Table 1).

Type	**Entries**	**%**
Underspecified	1012	49.1
Semantic information without a single concept	66	3.2
Single concept	598	29.0
Multiple concepts	386	18.7
Total	**2062**	**100**

Table 1: Semantic information for the *Sun Up to Sun Down* text

This leads directly to another set of problems. In a few cases, the semantic information in the knowledge base showed inconsistencies, such as reversed argument structures, wrong frame keywords, and incorrect part of speech information. These inconsistencies are rare and easy to correct, but they are difficult to detect in advance.

The interpretation process can also fail when the general semantic interpretation data contains expressions that are not recognized by the frame building rules as relevant for the instantiation of a particular QP frame. In such cases, new rules have to be added to the existing set.

Conclusions

As the example has illustrated, the correspondences between natural language and QP theory can be used to extract relevant information from simple paragraph-sized descriptions of instances of physical processes. QP theory is used in the interpretation to capture information about the underlying processes. The frame data is comparable to the information for the QP constituents that one would expect in manually constructed models.

The system makes use of the correspondences between natural language and QP theory in form of grammatical and interpretation rules. The results of this analysis were used in the development of our controlled language for the natural language input descriptions and the interpretation rules for in the generation of QP frame structures. It is important to note that the controlled language and the interpretation rules are domain-independent and intended to be applicable to any scenario that fits within the framework of QP theory.

Traditionally, models have been built by hand, based on a detailed analysis of the domain and requiring in-depth knowledge of the modeling language (see Kuipers & Kassirer (1984) for an example). Our goal is to allow domain experts specify their models as natural language text descriptions. Although the interpretation process requires more explicit and detailed descriptions than those usually found in unrestricted natural language text, we believe that the information captured as QP frames can be used as first cut representation to build more complete models through further refinement.

The next steps of our work will include such refinements of the semantic interpretation process and its rules and syntactic patterns, as well as a more flexible controlled language for descriptions. We are currently designing a module that translates the captured information (QP frames and scenario data) into model fragments. Furthermore, we plan to move beyond descriptions of instances of processes by including general information about physical phenomena. This information can come from two different sources: controlled language descriptions, similar to those of concrete instances, and generalized knowledge acquired from instances through an abstraction process (Kuehne, Forbus, Gentner, & Quinn, 2000).

Another important aspect is the availability of additional semantic information. As the previous section has shown,

[6] These numbers are based on Cyc knowledge base version 576, October 2003

[7] The list of words included the root lexicon entries, i.e. no inflected forms, for every available part of speech.

less than half the words in the COMLEX lexicon are covered by the knowledge base, and semantic information exists for only a fraction of them. Since descriptions of physical processes often contain domain-specific terminology, the lack of semantic information becomes an increasing problem. Although outside the scope of our current project, addressing these knowledge engineering issues is an important step towards the use of large knowledge bases for deep semantic NLP.

Acknowledgements

We like to thank Dedre Gentner, Chris Riesbeck, and Larry Birnbaum for their invaluable insight, comments and suggestions. This research was supported by the Artificial Intelligence program of the Office of Naval Research.

References

Allen, J. F. (1995). *Natural Language Understanding* (2nd ed.). Redwood City, CA: Benjamin/Cummings.

Almquist, I., & Sagvall Hein, A. (1996). *Defining Scania Swedish - a Controlled Language for Truck Maintenance.* In Proceedings: First International Workshop on Controlled Language Applications (CLAW-96), University of Leuven, Belgium.

Baker, C. F., Fillmore, C. J., & Lowe, J. B. (1998). *The Berkeley FrameNet Project.* In Proceedings: 17th International Conference on Computational Linguistics and 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 98), Montreal, Canada.

Barker, K., Delisle, S., & Szpakowicz, S. (1998). *Test-driving TANKA: Evaluating a Semi-Automatic System of Text Analysis for Knowledge Acquisition.* In Proceedings: Canadian Conference on Artificial Intelligence.

Buckley, S. (1979). *From Sun Up to Sun Down*. New York: McGraw-Hill.

Falkenhainer, B., Farquhar, A., Bobrow, D., Fikes, R., Forbus, K., Gruber, T., et al. (1994). *CML: A Compositional Modeling Language* (Technical Report KSL-94-16). Stanford, CA: Stanford University, Knowledge Systems Laboratory.

Fillmore, C. J., Wooters, C., & Baker, C. F. (2001). *Building a Large Lexical Databank Which Provides Deep Semantics.* In Proceedings: Pacific Asian Conference on Language, Information, and Computation, Hong Kong, China.

Forbus, K. D. (1984). Qualitative Process Theory. *Artificial Intelligence, 24*, 85-168.

Forbus, K. D. (1996). Qualitative Reasoning. In *CRC Handbook of Computer Science and Engineering* (pp. 715-733): CRC Press.

Forbus, K. D., & de Kleer, J. (1993). *Building Problem Solvers*. Cambridge, MA: MIT Press.

Fuchs, N. E., & Schwitter, R. (1996). *Attempto Controlled English (ACE).* In Proceedings: First International Workshop on Controlled Language Applications (CLAW-96), University of Leuven, Belgium.

Kuehne, S. E. (2003). *On the Representation of Physical Quantities in Natural Language.* In Proceedings: Seventeenth International Workshop on Qualitative Reasoning (QR '03), Brasilia, Brazil.

Kuehne, S. E. (2004). *Understanding Natural Language Descriptions of Physical Phenomena.* Ph.D. thesis, Northwestern University, Evanston, IL.

Kuehne, S. E., & Forbus, K. D. (2002). *Qualitative Physics as a component in natural language semantics: A progress report.* In Proceedings: Twenty-fourth Annual Conference of the Cognitive Science Society, George Mason University, Fairfax, VA.

Kuehne, S. E., Forbus, K. D., Gentner, D., & Quinn, B. (2000). *SEQL: Category Learning as Progressive Abstraction using Structure Mapping.* In Proceedings: Twenty-second Annual Conference of the Cognitive Science Society, Institute for Research in Cognitive Science, Philadelphia, PA.

Kuipers, B. J., & Kassirer, J. P. (1984). Causal reasoning in medicine: Analysis of a protocol. *Cognitive Science, 8*, 363-385.

Lenat, D. B., & Guha, R. V. (1989). *Building large knowledge-based systems : representation and inference in the Cyc project*. Reading, MA: Addison-Wesley.

Macleod, C., Grishman, R., & Meyers, A. (1998). *COMLEX Syntax Reference Manual, Version 3.0*. Philadelphia, PA: Linguistic Data Consortium, University of Pennsylvania.

Mahesh, K., & Nirenburg, S. (1995). *A Situated Ontology for Practical NLP.* In Proceedings: IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal.

Mitamura, T., & Nyberg, E. H. (1995). *Controlled English for Knowledge-Based MT: Experience with the KANT System.* In Proceedings: 6th International Conference on Theoretical and Methodological Issues in Machine Translation, Leuven, Belgium.

Nyberg, E. H., & Mitamura, T. (1992). *The KANT System: Fast, accurate, high-quality translation in practical domains.* In Proceedings: 15th International Conference on Computational Linguistics (COLING 92), Nantes, France.

Ogden, C. K. (1933). *Basic by Examples*. London: K. Paul, Trench, Trubner and Co., Ltd.

Wisniewski, E. J., & Love, B. C. (1998). Properties and relations in conceptual combination. *Journal of Memory and Language, 38*, 177-202.

Wojcik, R. H., Holmback, H., & Hoard, J. (1998). *Boeing Technical English: An Extension of AECMA SE beyond the Aircraft Maintenance Domain.* In Proceedings: Second International Workshop on Controlled Language Applications (CLAW 98), Pittsburgh, PA.