Boeing's NLP System and the Challenges of Semantic Representation

Peter Clark
Phil Harrison
The Boeing Company (USA)
email: `peter.e.clark@boeing.com`

Abstract

We describe Boeing's NLP system, BLUE, comprising a pipeline of a parser, a logical form (LF) generator, an initial logic generator, and further processing modules. The initial logic generator produces logic whose structure closely mirrors the structure of the original text. The subsequent processing modules then perform, with somewhat limited scope, additional transformations to convert this into a more usable representation with respect to a specific target ontology, better able to support inference. Generating a semantic representation is challenging, due to the wide variety of semantic phenomena which can occur in text. We identify seventeen such phenomena which occurred in the STEP 2008 "shared task" texts, comment on BLUE's ability to handle them or otherwise, and discuss the more general question of what exactly constitutes a "semantic representation", arguing that a spectrum of interpretations exist.

1 System Description

1.1 Overview and Scope

As our contribution to the 2008 STEP Symposium's "shared task" of comparing semantic representations (Bos, 2008), we describe Boeing's NLP system, BLUE (Boeing Language Understanding Engine), and subsequently analyze its performance on the task's shared texts. BLUE consists of a pipeline of a parser, logical form (LF) generator, an initial logic generator, and subsequent processing modules. The parser has broad coverage and is domain general. The logical form generator currently deals with a (reasonably large) subset of linguistic phenomena, including simple sentences, prepositional phrases, compound nouns, ordinal modifiers, proper nouns, some simple types of coordination, adverbs, negation, comparatives, and modals. The initial logic generator performs a straightforward transformation of the LF to first-order logic syntax. Subsequent processing modules then perform word sense disambiguation, semantic role labeling, coreference resolution, and some limited metonymic and other transformations. The overall system currently produces output expressed in one of two target ontologies, namely WordNet and the University of Texas at Austin's Component Library (CLib) (Barker et al., 2001). In this paper we illustrate the system's use with WordNet's ontology. The overall system was originally developed for interpreting a controlled language called CPL (Clark et al., 2007), but also often makes reasonable interpretations of more complex, open text sentences, as we illustrate here.

1.2 Parsing and the Logical Form Generator

Parsing is performed using SAPIR, a mature, bottom-up, broad coverage chart parser (Harrison and Maxwell, 1986). The parser's cost function is biased by a database of manually and corpus-derived "tuples" (good parse fragments), as well as hand-coded preference rules. During parsing, the system also generates a logical form (LF), a semi-formal structure between a parse and full logic. The LF is a simplified and normalized tree structure with logic-type elements, generated by rules parallel to the grammar rules, that contains variables (prefixed by underscores "_") and additional expressions for other sentence constituents. Variables can represent noun phrases, propositions, and even verb phrases (e.g., "To solve this problem is difficult").

Some disambiguation decisions are performed at this stage (e.g., structural, part of speech), while others are deferred (e.g., word senses, semantic roles), and there is no explicit quantifier scoping. Various syntactic properties and relationships are captured in the LF, including: S (sentence), PP (prepositional phrase), NN (noun compound), PN (proper name), PLUR (plural), PLUR-N (numbered plural). Tense, aspect, and polarity are also recorded in the LF. For example:

```
;;; LF for "An object is thrown with a horizontal speed of 20 m/s
          from a cliff that is 125 m high."
(DECL
   ((VAR _X1 "an" "object")
    (VAR _X3 NIL (PLUR-N "20" "m/s"))
    (VAR _X2 "a" "horizontal speed" (PP "of" _X3))
    (VAR _X4 "a" "cliff"
      (DECL NIL (S (PRESENT) _X4 "be"
        (S-ADJ _X4 (DEGREE
              (MEASUREMENT "125" "m") "high"))))))
   (S (PRESENT) NIL "throw" _X1
(PP "with" _X2) (PP "from" _X4)))
```

1.3 The Initial Logic Generator

The LF is then used to generate ground logical assertions of the form r(x,y), containing Skolem instances (denoting existentially quantified variables) by applying a set of simple, syntactic rewrite rules recursively to it. Verbs are reified as individuals, Davidsonian-style. At this stage of processing, the binary predicates are: subject (syntactic subject), sobject (syntactic object), mod (modifier), all the prepositions, value (for physical quantities), number-of-elements (for numbered plurals), and named (for proper names). For example, the above LF is translated into "syntactic logic" (additional predicates indicating part of speech, tense, aspect, determiners, and polarity are not shown):

```
;;; "An object is thrown with a horizontal speed
;;;     of 20 m/s from a cliff that is 125 m high."
"object"(object01),
value(quantity01,[20,m/s_n1]),
"m/s"(m/s_n1),
"speed"(speed01),
"horizontal"(horizontal01),
mod(speed01,horizontal01),
"of"(speed01,quantity01),
"cliff"(cliff01),
"be"(be01),
subject(be01,cliff01),
sobject(be01,height01),
value(height01,[125,m_n1]),
"m"(m_n1),
"height"(height01),
"throw"(throw01),
sobject(throw01,object01),
"with"(throw01,speed01),
"from"(throw01,cliff01).
```

1.4 Subsequent Processing Modules

While the output of the basic system is in a logic syntax, it is not coherent enough to support inference as it preserves many difficult linguistic phenomena (ambiguity,

metonymy, etc.). Further semantic interpretation involves disambiguation and aligning the interpretation with the target ontology we are using. In general, this is a complex task and our system only makes limited steps in this direction using five modules: word sense disambiguation (WSD); semantic role labeling (SRL); coreference resolution (including across different parts of speech); metonymy resolution (with respect to the target ontology); and structural transformations. We describe these modules below.

Word Sense Disambiguation   When using WordNet's ontology, each synset in Word-Net is a target concept for WSD. BLUE currently performs naive word sense disambiguation by simply selecting the most common synset for a given word+part-of-speech using context-independent frequency statistics. When using the Component Library (CLib) ontology, BLUE exploits hand-authored mappings between WordNet synsets and CLib concepts: Given a word, e.g., "cliff", BLUE first finds WordNet synsets for the word, then climbs WordNet's taxonomy from those synsets until it finds synsets mapped to CLib concepts, and returns those CLib concepts, again using preference based on context-independent frequency statistics. Verb nominalizations map to the denominalized verb, thus "fall"(n) and "falling"(n) both map to synsets for "fall"(v).

Semantic Role Labeling   With both ontologies, BLUE uses the same relational vocabulary of approximately 100 binary semantic relations, drawn from the relation set used by UT's Component Library. Semantic role labeling (SRL), for both for verb-noun and noun-noun relationships, is performed using a set of hand-authored SRL rules, e.g., "from"(x,y) is labeled as origin(x,y) if x is a movement event and y is an object. In cases where the rules are not adequate to clearly identify a semantic relation, the relation is left as a syntactic relation.

Coreference   Coreference (e.g., "the ball") is computed by searching for a previous entity in the discourse with the same word and qualifiers as in the referring noun phrase. (Coreference using synonyms or types produced more errors than it removed).

Metonymy   Often a sentence relates entities in a way inconsistent with the target ontology. For example, with the Component Library (CLib) ontology, movement properties (e.g., speed, acceleration) are defined as properties of the movement events, rather than of the object moving. Thus a phrase like "the initial speed of the ball" is metonymous (with respect to CLib) for "the initial speed of the movement of the ball". This module spots and corrects such metonymies using a small set of metonymy resolution rules. Note that metonymy resolution is ontology-specific, reflecting design decisions about what is and is not an allowable expression in the target ontology.

Structural Transformations   Often, the structure of the syntactic and (desired) semantic representations differ, and so some structural transformations are necessary. For example, in the basic processing, verbs (e.g., "weigh") are reified as individuals with semantic roles, e.g., "weigh"(w), subject(w,x), sobject(w,y), whereas the target ontology stipulates that some particular verbs denote relations e.g., "weigh" corresponds to the CLib relation weight(x,y), not a Weigh event. (This is indicated in CLib by the relation weigh() being associated with synsets for the verb "weigh"). Similarly, nouns associated with relations will be transformed to introduce that relation into the representation, e.g., "weight"(y), "of"(y,x) will be transformed to weight(x,y). This

module makes these and other transformations. The verbs "be" and "have" are similarly mapped to relations, but with the extra step that the target relation depends on the arguments. A small set of rules determines the appropriate relation to use.

2 Semantic Formalism

2.1 Form (Syntax)

Our system produces output in a subset of first-order logic, illustrated later in this paper. For the most part, it simply outputs a flat list of ground assertions containing Skolemized existential variables, and does not handle universal quantification (a significant limitation for expository rather than story-like texts). In addition, BLUE allows propositions to themselves be arguments to other propositions as a nested structure, e.g., for modals:

```
;;; "The man wanted to leave the house"
isa(leave01,leave_v1), etc, ...
agent(want01,man01),
object(want01,[
        agent(leave01,man01),
        object(leave01,house01)]).
```

2.2 Ontology (Content)

As described earlier, BLUE currently uses two alternative conceptual vocabularies, namely the concepts in WordNet (with minor extensions) or the Component Library. BLUE's relational vocabulary is approximately 100 semantic relations drawn from the Component Library.[1]

3 Example

We illustrate our system using an example from Project Halo (Clark et al., 2007), where the system is used to interpret multi-sentence science questions posed to a knowledge-based system. While BLUE produces a slightly better output for this text using the Component Library ontology, we illustrate it using WordNet's ontology for consistency with our output for the other shared task texts (we use WordNet for these as WordNet has broader coverage). We also discuss our system further in Section 4 on additional sentences.

The first three sentences are (largely) a question from an AP Physics exam, the fourth is a hand-written simplified version of the third sentence. Our system is able to create coherent representations of sentences 1, 2, and 4, i.e., sufficient for the KB to answer the question correctly, but not of sentence 3.

Shared Task Text 1:
(1.1) An object is thrown with a horizontal speed of 20 m/s from a cliff that is 125 m high.
(1.2) The object falls for the height of the cliff.
(1.3) If air resistance is negligible, how long does it take the object to fall

---

[1]http://www.cs.utexas.edu/users/mfkb/RKF/trunktree/components/specs/
slotdictionary.html

to the ground?
(1.4) What is the duration of the fall?

Semantic Representation

```
(1.1) "An object is thrown with a horizontal speed of 20 m/s
       from a cliff that is 125 m high."
isa(object01,object_n1),
isa(speed01,velocity_n1),
isa(horizontal01,horizontal_a1),
isa(cliff01,cliff_n1),
isa(height01,height_n1),
isa(throw01,throw_v1),
height(cliff01,height01),
value(speed01,[20,m/s_n1]),
mod(speed01,horizontal01),
value(height01,[125,m_n1]),
object(throw01,object01),
"with"(throw01,speed01),
origin(throw01,cliff01).
```

Here object01 etc. denote Skolem instances, object_n1 etc. denote WordNet concepts (synsets). Note word and role disambiguation, adjective-noun transformation ("high" → height()), "be" interpretation, and handling of units of measurement ("125 m", "20 m/s"). Using WordNet's ontology, this interpretation is not perfect as two semantic roles have (undesirably) been left underspecified ("mod" and "with").

```
(1.2) "The object falls for the height of the cliff."
isa(fall01,fall_v1),
height(cliff01,height01),
agent(fall01,object01),
distance(fall01,height01).
```

Note coreference with first sentence ("height", "cliff", "object") and semantic role labeling ("for height" → height()).

```
(1.3) "If air resistance is negligible, how long does it
       take the object to fall to the ground?"
```

(See the STEP Shared Task Web site[2] for BLUE's semantic representation). BLUE's representation for this is largely incoherent, in particular a "take" event is created with a proposition (meaning "length of fall to the ground") as its 2nd argument.

```
(1.4) "What is the duration of the fall?"
isa(fall01,fall_v1),
isa(duration01,duration_n1),
duration(fall01,duration01),
query-for(duration01).
```

Note noun-verb coreference ("fall"(n) → fall01) and query variable identification.

---

[2]http://www.sigsem.org

4   Performance on All Shared Texts

As part of the STEP 2008 Symposium, seven groups (including us) each submitted a paragraph of text and then all groups ran their NLP systems on all texts (Bos, 2008). We now discuss BLUE's capabilities further in the context of these shared texts. For this exercise, we made some minor bug fixes to the system but did not significantly change or extend the final output representations. In the below discussion we refer to the text and sentence numbers in the form (text#.sentence#). Sometimes text snippets have been simplified for clarity.

What constitutes a Semantic Representation?

The notion of a semantic representation can be interpreted in several ways. At one extreme, a representation which captures all the salient linguistic structure and phenomena could be considered "semantic". Such representations will have structure somewhat similar to the syntactic structure of the original text, and the task of interpreting the inferential consequences of those structures is then left to downstream processing, and considered part of commonsense reasoning rather than "language understanding". At the other extreme, one might require the full logical interpretation of the text to be explicit, in order that the representation be truely "semantic", on the grounds that if the representation does not explicitly support inference of valid consequences, the meaning has not been captured. Various positions exist between these two extremes. For example, one might represent:

>   (4.2) "a vaccine prevents cervical cancer"

as

a. prevents(vaccine,cervical-cancer); or

b. $\exists v$ type-of(v,vaccine) & $\forall x,y$ isa(x,v), isa(y, cervical-cancer) $\rightarrow$ prevents(x,y); or

c. the logic for, approximately, "for all people given (a specific type of) vaccine, they will not subsequently develop cervical cancer"

>   Similarly, one might represent "typical" in

>   (7.4) "turbines had a typical power rating of 150 kW."

as

a. have(turbine,power-rating), value(power-rating, 150kW), typical(power-rating); or

b. logic for (say) "the mode of the power rating of the set of turbines is (approximately) 150kW"

Clearly the more a representation is syntax-like, the more a downstream reasoning component will need have to do to identify its inferential consequences. Conversely, the more a representation makes the meaning explicit, the harder it is to generate those representations in the first place as even simple sentences can often have highly complex meanings. At what point one considers a representation "semantic" is matter

for debate; what is clear is that there is often a significant journey to make to get from text to valid inferential consequences of that text. From a pragmatic point of view, like most other language systems BLUE generates representations which are more syntactically structured. This means that, whether one considers them "semantic" or not, considerable additional machinery would typically be needed for performing inference using them.

Some other examples of simple sentences with complex meanings include:

- (6.2) "selling a range of produce" meaning, approximately, "the number of types of produce sold is reasonably large";

- (6.4) "research has fluctuated with tax incentives" meaning, approximately, a qualitative relationship exists between the amount of research and the amount of tax incentives;

- (7.2) "electricity distribution spread to farms" appealing to the abstract notion of a spatial region, and meaning, approximately, that the region grows with time.

Even with a more syntactic notion of "semantic representation", there are numerous more specific issues which need to be addressed. Below we identify some which arise in the shared task texts, and comment on our system's ability to handle them.

4.1   Word Sense Disambiguation (WSD)

BLUE currently uses a naive, context-independent approach to WSD. While the naive guess is often right, there are many cases in the shared texts of unusual senses which BLUE will miss, e.g., (1.3) "how long [time] does it take", (2.2) "led to [inspired the development of] a vaccine", (6.2) "turn [generate] a profit".

An interesting phenomenon is seen with: (6.2) "Greensgrow [is] a plot of land and is selling its own vegetables". which mixes senses of "Greensgrow" as a piece of land and an institution in the same sentence, causing challenges for standard WSD. One might consider "Greensgrow" as denoting an institution and thus "Greensgrow is a plot of land" as metonymy, or "Greensgrow" as a complex concept with various facets. In either case some complex processing is required.

4.2   Semantic Role Labelling (SRL)

SRL is itself challenging. In many cases BLUE has left the relation underspecified (especially noun-noun relations), and has occasionally maked mistakes, e.g.,

```
(3.2) "A table in the corner"
is-inside(table01, corner01).

(5.1) "I have problem"
has-part(i01,problem01).
```

4.3   Coordination

BLUE will multiply out coordinates, e.g.:

```
(3.4) "The atmosphere was warm and friendly"
"be"(atmosphere01, warm01).
"be"(atmosphere01, friendly01).
```

Sometimes this multiplication is inappropriate, for example below, BLUE misinterprets each place as being in both England and France simultaneously:

```
(4.3) "They visited places in England and France"
object(visit01,place01),
is-inside(place01,England01),
is-inside(place01,France01).
```

Note that the alternative "places in Africa and South of the Equator" would not be inconsistent as these areas do overlap; domain knowledge is thus required to understand the intended semantics.

BLUE does not distribute modifiers across coordinates, and thus misses the distribution (7.1) "wind-energy technology and applications" → "wind-energy technology and wind-energy applications".

4.4 Coreference and Anaphora

BLUE performs definite reference resolution based on name, e.g., (1.2) "an object...the object...", and across part of speech, e.g., verb-noun (1.4) "falls... the fall...", and adjective-noun (1.2) "high... height...", but not across different names, e.g., (6.3) "Greenslow... The farm...". BLUE does not currently do anaphoric reference resolution, so leaves occurrences of "it" etc. unresolved.

There are some interesting complex examples of coreference also in the texts:

- (3.3) "The waiter took the order"
  The two referents are not mentioned earlier, but are understood by the reader to refer to objects in the described scene. A system should thus realize that "The waiter" is the waiter in the restaurant, and "the order" is John's order.

- (2.2) "Cervical cancer is caused by a virus. That has been known..."
  Here the anaphor ("That") refers to a proposition rather than an object in the world.

- (2.3) "other cancers"
  This refers to the set of cancers except those previously mentioned, requiring discourse analysis to fully capture the semantics.

4.5 Generics and Universal Quantification

BLUE interprets generics as statements relating individuals, thus requiring further downstream interpretation of those individuals and transformation of the representation (e.g., to universal quantification and conditionals) for correct inference. In general, generics are complex to interpret; not only are quantifications ambiguous, but also generics typically require substantial unstated information to be filled in for the interpretation to be meaningful. For example,

(2.1) "Cervical cancer is caused by a virus"

should ultimately be interpreted as (something like) "An event involving a virus can create an incidence of cervical cancer". An even fuller semantics, requiring more world knowledge, would be that the event is infection of a person and that the cancer incidence is in the same person. How far one should go to reach this degree of interpretation in a "semantic representation" is open to debate.

Adjectives and adverbs can modify the expectation of measurement results on an ensemble, again requiring special representational machinery. Examples include:

(7.4) "turbines have a typical power rating of..."
(7.5) "turbines are commonly rated as"
(6.1) "the community often lacks it [fresh food]"

4.6 Time

BLUE ignores tense and aspect information (although it extracts it in the intermediate logical form), a gap for complete semantics. However, BLUE will handle some references to events situated in time and in relation to other events. In the examples below of BLUE's interpretation, note the use of temporal predicates:

```
(5.5) "...made in 1945"
time-int-during(make01,year1945)

(4.4) "ensures operation until 1999"
time-ends(ensure01, year1999).

(5.3) "...yelled. Then the propellant exploded"
next-event(yell01,explode01).

(5.4) "When they killed, they were crouching"
time-at(crouch01,kill01).
```

However BLUE does not recognize more complex time references as such, e.g.,

(7.1) "the 1930s",
(7.2) "the early 1970s",
(7.4) "mid-'80s" (misparsed as an adjective), and
(7.3) "the past 30 years"

In addition, facts or beliefs may themselves be situated in time, requiring a time-stamp or situation to be attached to an assertion (which BLUE does not do), for example (6.3) "revenue of $450,000 in 2007" A particularly complex example is (5.6) "Initially it was suspected that..." meaning, approximately, X suspected Y at the time immediately after the previously described event. Computationally disentangling the meaning of this sentence is a formidable challenge.

4.7 Plurals and Collectives

BLUE represents a numbered collective as an individual with a number-of-elements() predicate attached, for example:

```
(4.2) "seven people developed"
isa(person01, person_n1),
agent(develop01, person01),
number-of-elements(person01,7).
```

where person01 denotes the collective of 7 people. (While it is strictly incorrect to assert person01 as an instance of the class person_n1, there are pragmatic benefits for doing so.) Unnumbered plurals and generics are naively represented as single individuals at present.

4.8 "Light" nouns and verbs

Arguably, some nouns and verbs do not denote objects and events in the world in a literal sense. For example, "X occurred" can be taken to mean just "X", rather than there being a separate "occur" event. Recognizing and transforming these requires special processing machinery. BLUE handles a few examples, e.g., "X occurred" → "X", but not the following in the shared texts:

- (1.3) "how long does it [the fall] take" meaning "how [temporally] long is the fall"

- (4.3) "doing internship" meaning "interning"

- (5.4) "an explosion happened" meaning "There was an explosion"

- (7.1) "development was underway" meaning "There was development"

4.9 Adjectives and Adverbs

While an adjective or adverb can be trivially attached to a noun/verb, as BLUE does, an elaborated representation of its meaning, as required for inference, is very challenging and context-dependent. Challenging examples include:

- (6.1) "North Philadelphia"; what is the extent of this region?

- (7.1) "modern development"

- (4.3) "similar places"; other entities with properties close to some currently mentioned entity

- (4.3) "future trainer"; a non-intersective adjective (like "fake gun")

- (5.4) "crouching unnaturally"

4.10 Modals and Higher-Order Expressions

BLUE will handle some modal expressions by placing a proposition as an argument to another proposition, for example:

```
(4.5) "We would like our school to work similarly..."
agent(like01,we01).
object(like01,[agent(work01,school01),
               manner(work01,similarly01)])
```

```
(5.6) "It was suspected that this storage reduced the
        powder's stability."
object(suspect01,["of"(stability01,powder01),
                    agent(reduce01,storage01),
                    object(reduce01,stability01)]).
```

A particularly complex example (which BLUE does not handle) is (5.3) "They were crouching, which suggested that they knew that an explosion would happen." where a past event implies belief in a future event's occurrence.

4.11   Uncertainty and possibility

The phrases (2.3) "cancers may be caused by viruses" and (5.6) "the storage might have reduced stability" have a complex semantics concerning possibility. While we represent the may/might aspect in the initial logical form, BLUE ignores it in the subsequent representations.

4.12   Metonymy

The occurrence of metonymy is somewhat subjective because metonymy is relative to a target ontology. A full semantic interpretation would include metonymy resolution where present. BLUE will resolve some special cases of metonymy, in particular with respect to the Component Library ontology, but those did not occur in these texts. Some metonymy-like examples in the shared texts (which BLUE did not resolve) include:

(6.3) "The [people of the] farm hopes to make a profit"
(2.1) ".cancer is caused by a virus [infection]."
(7.3) "[The amount of] research has fluctuated"

4.13   Implicit Arguments

Sometimes a verb or noun has implicit arguments that an interpretation should make explicit. BLUE will recognize some implicit arguments for modals, e.g.: that .the farm. is the implied object making the profit in the below:

```
(6.3) "The farm hopes to make a profit"
agent(hope01,farm01),
object(hope01,[ agent(make02,farm01),  ; implicit arg found
                object(make02,profit02)])
```

but not in other cases, such as in (6.3) "revenue of $450,000" → "the revenue of the farm was $450,000".

In general, many relationships are unstated in text and need to be inferred for a full understanding. Text 3 (the restaurant story) is particularly challenging in this regard.

4.14   Special Constructs

There are some specialized grammatical constructs which are not inherently complex to handle, but require special processing. Examples include money, e.g., (6.3) "$10,000", dates, e.g., (7.4) "mid-'80s", and units of measure, e.g., (1.1) "m/s". BLUE currently only recognizes the latter, which is hard-coded as a single token. BLUE also does not handle quote characters, e.g., (5.1) ... yelled "I have a problem" ...

4.15 Proper Names

BLUE will recognize proper names and encode them with a specific named() predicate, e.g.,:

```
(4.2) "Joao Pedro Fonseca"
isa(Fonseca01, person_n1),
named(Fonseca01, ["Joao","Pedro","Fonseca"]).
```

4.16 Physical quantities

Physical quantities need special processing. BLUE represents physical quantities using a special predicate (called `value()`) linking the quantity to its magnitude and unit of measurement, e.g.,:

```
(1.1) "125 m"
value(height01,[125,m_n1]).
```

4.17 Questions

BLUE recognizes several question types ("what is the...", "what is a...", "how many...", "how much...", "is it true that...", "why...", "how...") and represents them using special annotations on the variable/proposition in the query, for example:

```
(1.4) "What is the duration?"
query-for(duration01).
```

5   Summary and Conclusion

Our language system, BLUE, is able to generate representational structures for many texts, capturing numerous linguistic phenomena while also missing or misinterpreting a variety of others. We have presented a small catalog of these phenomena, and comments on BLUE's ability to handle them or otherwise. As discussed, BLUE's output representation is still fairly linguistic in structure, and despite some transformations would often require substantial downstream processing to identify the explicit meaning and inferential consequences of those structures. Despite this, for cases where the gap between syntax and final logical semantics is small, in particular for the controlled language subset it was originally designed to support, it can generate useful output, and thus constitutes a small step along the way to language understanding.

References

Barker, K., B. Porter, and P. Clark (2001). A library of generic concepts for composing knowledge bases. In *Proc. 1st Int Conf on Knowledge Capture (K-Cap'01)*, pp. 14–21. ACM.

Bos, J. (2008). Introduction to the Shared Task on Comparing Semantic Representations. In J. Bos and R. Delmonte (Eds.), *Semantics in Text Processing. STEP 2008 Conference Proceedings*, Volume 1 of *Research in Computational Semantics*, pp. 257–261. College Publications.

Clark, P., J. Chaw, J. Thompson, and P. Harrison (2007). Capturing and answering questions posed to a knowledge-based system. In D. Sleeman and K. Barker (Eds.), *Proc Int Conf on Knowledge Capture (KCap'07)*, pp. 63–70.

Clark, P., P. Harrison, J. Thompson, R. Wojcik, T. Jenkins, and D. Israel (2007). Reading to learn: An investigation into language understanding. In *Proc. AAAI Spring Symposium on Machine Reading*. AAAI.

Harrison, P. and M. Maxwell (1986). A new implementation of GPSG. In *Proc. 6th Canadian Conf on AI (CSCSI-86)*, pp. 78–83.