The TEXTCAP Semantic Interpreter

Charles B. Callaway

University of Edinburgh (UK)

email: ccallawa@inf.ed.ac.uk

Abstract

The lack of large amounts of readily available, explicitly represented knowledge has long been recognized as a barrier to applications requiring semantic knowledge such as machine translation and question answering. This problem is analogous to that facing machine translation decades ago, where one proposed solution was to use human translators to post-edit automatically produced, low quality translations rather than expect a computer to independently create high-quality translations. This paper describes an attempt at implementing a semantic parser that takes unrestricted English text, uses publically available computational linguistics tools and lexical resources and as output produces semantic triples which can be used in a variety of tasks such as generating knowledge bases, providing raw material for question answering systems, or creating RDF structures. We describe the TEXTCAP system, detail the semantic triple representation it produces, illustrate step by step how TEXTCAP processes a short text, and use its results on unseen texts to discuss the amount of post-editing that might be realistically required.

1 Introduction

A number of applications depend on explicitly represented knowledge to perform basic tasks or add customization to existing tasks. Improving the quantity and quality of the knowledge contained in knowledge bases could lead to the improved performance of many applications that depend on knowledge and inference such as:

- Generating scientific or educational explanations of natural or mechanical systems and phenomena (Lester and Porter, 1997),
- Question answering systems (Clark et al., 2001) that use reasoning to solve problems rather than looking up answers,
- Multimodal information presentation systems that depend on specific real world knowledge in order to describe or refer to it for audiences (Callaway et al., 2005; Stock et al., 2007).

These systems have typically relied on hand-built and domain specific knowledge bases requiring years of effort to produce. The need to speed up this process as well as make the resulting representations more consistent are well-known problems that have yielded a number of potential solutions (Blythe et al., 2001; Reiter et al., 2003; Carenini et al., 2005; Barker et al., 2007), but large scale, domain independent, and fully automatic knowledge acquisition on unrestricted text is still in its infancy.

Over the last decade research in applied computational linguistics has extended the various components necessary for semantic parsing, but have tended to focus on increasing the measurable performance of individual subtask in isolation (e.g., parsing, anaphora resolution, semantic role labelling, and word sense disambiguation) rather than on an entire end-to-end system. Meanwhile, theoretical CL research has examined issues such as underspecification, scoping and reference resolution in discourse contexts, but has set aside issues such as large-scale robustness, ontology integration and evaluation which are vital for applied uses of semantic parsing.

In this paper we discuss an implementation to automatically extract explicitly coded conceptual and ontological knowledge from unrestricted text using a pipeline of NLP components, as part of the STEP shared task (Bos, 2008). The TEXTCAP system performs the basic steps towards this task by gluing together an off-the-shelf parser with semantic interpretation methods. It is intended to be a test case for (1) establishing baseline performance measures for semantic parsing and (2) determining what degree of post-editing might be necessary in real-world environments.

Because major components of such a system would not be tailored towards the semantic parsing task, we would rightly expect its output to be imperfect. This problem is analogous to that facing machine translation decades ago, where one proposed solution was to use human translators to post-edit automatically produced, low quality translations rather than expect a computer to independently create high-quality translations. One aspect of this research is thus to investigate how much post-editing would be required to convert the system's output to usable semantic triples.

Finally, this paper presents the results of TEXTCAP on the 2008 STEP shared task corpus, giving specific comments about the difficulties in encountered. Although not a formal evaluation, we were satisfied with its performance in terms of accuracy and efficiency for helping humans post-edit semantic triples.

2 System Description

TEXTCAP performs basic steps towards the task of converting free text into semantic triples by gluing together an off-the-shelf parser with ad-hoc semantic interpretation methods. TEXTCAP parses a document into Penn TreeBank form and then traverses each syntactic parse tree performing a series of step-by-step tasks such as discourse parsing, clause separation, word sense disambiguation, anaphora resolution and semantic role labelling. Ad hoc rules then create a set of triples from the resulting semantically-enhanced parse tree.

TEXTCAP first uses the domain-independent Charniak parser (Charniak, 2000) to convert sentences in the source document into a sequence of syntactic parses. It then applies syntax-based discourse parsing rules (such as Soricut and Marcu (2003)) to reduce coordinate, subordinate, and relative clauses into coindexed, simpler sentence parses headed by single verbal relations.

It then marks for grammatical roles (subject, object, etc.) and syntactic features (e.g., passivity) before using a simple anaphora resolution algorithm based on those features and a word sense disambiguation algorithm grounded in WordNet (Fellbaum, 1998) senses that helps determine additional features such as animacy. A two-pass method is applied where first monosemous words are assigned senses, and then remaining senses are selected together with verb types (TEXTCAP uses ad hoc rules rather than current verb taxonomies like FrameNet). Selectional restrictions from the verb type then allows for labelling of peripheral grammatical roles as semantic roles. Finally, entities representing specific objects are marked with ontological relations and discourse relations are realized between individual verbal relations.

The end product of TEXTCAP is thus a list of coindexed semantic triples representing the explicitly recoverable semantic content of the input text.

3 Text Processing Components

Corpus methods underlie many of the recent improvements in a wide array of generic NLP tools. For instance, the introduction of large-scale lexical and syntactic resources like the Penn TreeBank (Marcus et al., 1993) have led to highly accurate, domain independent parsers (Collins, 1999; Charniak, 2000). Wide-coverage anaphora resolution systems process references across multiple sentences, and recent work on anaphora resolution by Poesio and Kabadjov (2004) describes itself as the first such system which can be used off-the-shelf.

Word sense disambiguation (Gliozzo et al., 2005), often based on term frequency analyses of large annotated corpora, can help localize search in a particular area of a knowledge base to find the most related concepts and instances. Semantic role labelers (Gildea and Jurafsky, 2002; Yeh et al., 2006) annotate what role each entity has in relation to its local man verb, and can provide additional clues for disambiguating words and locating them in an ontological space.

In addition to lexical and semantic tasks, multi-sentence linguistic analysis such as discourse segmentation and parsing is needed to semantically label the roles of verb phrases in relation to one other. Soricut and Marcu (2003) presented a statistical system that automatically produces an analysis of the rhetorical structure that holds between sets of sentences or clauses at the paragraph level.

As a generalization, NLP research has been conducted separately and few attempts have been made to connect each of them into the longer chains and pipelines needed for more complete and deeper text processing such as is needed for tasks like knowledge acquisition. Additionally, most of these tools are intended to iteratively examine each sentence individually within a larger document. But often important linguistic phenomena cross sentence boundaries, yet are just as necessary to properly understand the semantic content of a document.

4 Knowledge Representation in TEXTCAP

A common method of representing semantic knowledge in the Knowledge Base community (Brachman and Schmolze, 1985; Clark and Porter, 1998) is through three-place predicates, or triples, of the form (CONCEPT RELATION CONCEPT). A concept can signify either a generic entity or class, like "houses", or a particular instance, such as "my house at 35 Lincoln Avenue"; instances are coindexed to indicate they are the same entity in multiple predicates. Relations are typed according to domain, range and cardinality, and can also be marked as instances to indicate that they refer to specific events or properties that hold at a particular time place, etc.

Databases and knowledge bases can both be represented as large collections of triples. Knowledge bases differ from databases in that they are generally organized around a hierarchical taxonomy, or ontology, of both entities and relations, allowing for subsumption as inference and for knowledge to be separated into subgroups. Knowledge bases differ from ontologies in that, like databases, they also contain a larger set of specific knowledge (instances) that describes non-taxonomic relationships between members and instances of the ontology's concepts.

For instance, the sentence "My dog chases rabbits." talks about a specific instance of the class `dog` and its relationship to the generic class representing `rabbits`, perhaps represented as the triple `(DOG492 CHASING RABBIT-ANIMAL)`. To know that this dog really is a member of what we consider as the class of all dogs, we would need to add an ontological triple such as `(DOG492 INSTANCE-OF DOG)`. To represent the possessive grammatical relation in "my dog" we would need to agree on some particular person (an instance) to represent the speaker of the utterance `(PERSON142 INSTANCE-OF PERSON)` and then also add a relation to indicate possession `(DOG492 OWNED-BY PERSON142)`. Because language is ambiguous compared to semantic triples, we wouldn't want the word "my" to always be mapped to the same relation, for instance, obtaining `(PERSON366 OWNED-BY PERSON142)` from the phrase "my friend".

Like concepts, relations can also have instances since they can refer to particular events with particular modifiers. For instance, in the sentence "My dog quickly chased rabbits yesterday." we would need to change the relation `CHASING` from the triple above to `(DOG492 CHASING141 RABBIT-ANIMAL)` to indicate its modifiers, perhaps with `(CHASING141 SPEED QUICKLY)` and `(CHASING141 EVENT-TIME YESTERDAY)`. We would also need to indicate the taxonomic relationship between the two relations, `(CHASING141 INSTANCE-OF CHASING)`.

Because over the years different research groups have created differing ontologies, it is important to have a common ontology (and arguably, mapping of lexical items to classes in that ontology) for purposes such as evaluative comparison, even if implementations that acquire semantic triples can use any available ontology.

In keeping with the practice of much recent large-scale NLP, TEXTCAP uses WordNet (Fellbaum, 1998) as an underlying ontology and sense repository for generic classes, giving it the ability to leverage recent NLP tools that rely on it, such as for word sense disambiguation (Gliozzo et al., 2005). Thus given the sentence in Figure 1(a), we are interested in producing the semantic triples in (b) where generic entities and relations are grounded in WordNet.

5 Processing The Text

To illustrate how TEXTCAP works, we follow how it processes the following paragraph of newspaper text from the New York Times:

> Amid the tightly packed row houses of North Philadelphia, a pioneering urban farm is providing fresh local food for a community that often lacks it, and making money in the process. Greensgrow, a one-acre plot of raised beds and greenhouses on the site of a former steel-galvanizing factory, is turning a profit by selling its own vegetables and herbs as well as a range of produce from local growers, and by running a nursery selling plants and seedlings. The farm earned about $10,000 on revenue of $450,000 in 2007, and hopes to make a profit of 5 percent on $650,000 in revenue in this, its 10th year, so it can open another operation elsewhere in Philadelphia.

The first sentence as parsed by Charniak and converted into Lisp notation is:

```
(S (PP (IN "Amid")
       (NP (NP (DT "the") (ADJP (RB "tightly") (VBN "packed"))
               (NN "row") (NNS "houses"))
           (PP (IN "of") (NP (NNP "North") (NNP "Philadelphia")))))
   (PUNCTUATION COMMA)
   (NP (DT "a") (JJ "pioneering") (JJ "urban") (NN "farm"))
   (VP (AUX "is")
       (VP (VP (VBG "providing")
               (NP (JJ "fresh") (JJ "local") (NN "food"))
               (PP (IN "for")
                   (NP (NP (DT "a") (NN "community"))
                       (SBAR (WHNP (WDT "that"))
                             (S (ADVP (RB "often"))
                                (VP (VBZ "lacks") (NP (PRP "it")))))))))
           (PUNCTUATION COMMA)
           (CC "and")
           (VP (VBG "making")
               (NP (NN "money"))
               (PP (IN "in") (NP (DT "the") (NN "process")))))))
```

```
(a)   "My dog quickly chased rabbits yesterday."

(b)   (DOG492 INSTANCE-OF DOG#n1)
      (PERSON142 INSTANCE-OF PERSON#n1)
      (CHASING141 INSTANCE-OF CHASING#v1)
      (DOG492 CHASING141 RABBIT#n1)
      (CHASING141 SPEED QUICKLY#adv1)
      (CHASING141 EVENT-TIME YESTERDAY#adv1)
```

Figure 1: WordNet senses as generic entities and relations

We first normalize this from the form used by the Charniak and Collins parsers (which do no semantic role labelling and introduce some simplifications) into a corrected version following the original Penn TreeBank format. In the above parse, the following lines are normalized to mark grammatical subject and correctly mark the auxiliary verb:

```
   . . .
   (NP-SBJ (DT "a") (JJ "pioneering") (JJ "urban") (NN "farm"))
   (VP (VBZ "is")
       (VP (VP (VBG "providing")
               . . .
```

We then apply a customized discourse parser which converts full syntactic parses into subparses headed by single verb relations. This is done using purely syntactic information to break up coordinate, subordinate and relative clauses while adding coindexed traces at the appropriate parse level and introducing a new tree-level tag DR for discourse relations marked according to Rhetorical Structure Theory (Mann and Thompson, 1987). All three sentences in the paragraph above are thus converted into the following 13 discourse parses:

```
(S (PP (IN "Amid")
       (NP (NP (DT "the") (ADJP (RB "tightly") (VBN "packed"))
           (NN "row") (NNS "houses"))
           (PP (IN "of") (NP (NNP "North") (NNP "Philadelphia")))))
   (PUNCTUATION COMMA)
   (NP-SBJ (DT "a") (JJ "pioneering") (JJ "urban") (NN "farm") (TRACE 1))
   (VP (VBZ "is")
       (VP (VP (VBG "providing")
               (NP (JJ "fresh") (JJ "local") (NN "food"))
               (PP (IN "for")
                   (NP (DT "a") (NN "community") (TRACE 2)))))))

(S (NP-SBJ (DT "a") (NN "community") (TRACE 2))
   (ADVP (RB "often"))
   (VP (VBZ "lacks") (NP (PRP "it"))))

(S (NP-SBJ (DT "a") (JJ "pioneering") (JJ "urban") (NN "farm") (TRACE 1))
   (VP (VBZ "is")
       (VP (VBG "making")
           (NP (NN "money"))
           (PP (IN "in") (NP (DT "the") (NN "process"))))))

(S (NP-SBJ (NNP "Greensgrow") (TRACE 3))
   (VP (VBZ "is")
       (NP (NP (DT "a") (JJ "one-acre") (NN "plot"))
           (PP (IN "of")
                (NP (NP (VBN "raised") (NNS "beds")
                        (CC "and") (NNS "greenhouses"))
                    (PP (IN "on")
                        (NP (NP (DT "the") (NN "site"))
                            (PP (IN "of")
                                (NP (DT "a") (JJ "former")
                                    (JJ "steel-galvanizing")
                                    (NN "factory"))))))))))

(S (NP-SBJ (NNP "Greensgrow") (TRACE 3))
   (VP (VBZ "is")
       (VP (VBG "turning")
           (NP (DT "a") (NN "profit"))))
   (TRACE 4))

(S (NP-SBJ (NNP "Greensgrow") (TRACE 3))
   (VP (VBZ "is")
       (VP (VBG "selling")
           (NP (NP (PRP-POSS "its") (JJ "own")
                   (NNS "vegetables") (CC "and") (NNS "herbs"))
               (CONJP (RB "as") (RB "well") (IN "as"))
               (NP (NP (DT "a") (NN "range"))
                   (PP (IN "of") (NP (NN "produce")))))
           (PP (IN "from") (NP (JJ "local") (NNS "growers")))))
   (TRACE 5))

(S (NP-SBJ (NNP "Greensgrow") (TRACE 3))
   (VP (VBZ "is")
       (VP (VBG "running")
           (NP (NP (DT "a") (NN "nursery")))))
   (TRACE 6))

(S (NP-SBJ (NNP "Greensgrow") (TRACE 3))
   (VP (VBZ "is")
       (VP (VBG "selling")
           (NP (NNS "plants") (CC "and") (NNS "seedlings")))))

(DR (MEANS (TRACE 4) (TRACE 5) (TRACE 6)))

(S (NP-SBJ (DT "The") (NN "farm") (TRACE 7))
   (VP (VBD "earned")
       (NP (QP (RB "about") (CURRENCY DOLLAR-SIGN) (CD 10000)))
       (PP (IN "on")
           (NP (NP (NN "revenue"))
               (PP (IN "of")
                   (NP (CURRENCY DOLLAR-SIGN) (CD 450000)))))
       (PP (IN "in") (NP (CD 2007)))))

(S (NP-SBJ (DT "The") (NN "farm") (TRACE 7))
   (VP (VBZ "hopes") (S (VP (TO "to") (VP (VBP "make")
          (NP (NP (NP (DT "a") (NN "profit"))
                  (PP (IN "of")
                      (NP (NP (CD 5) (NN "percent"))
                          (PP (IN "on")
                              (NP (NP (CURRENCY DOLLAR-SIGN) (CD 650000))
                                      (PP (IN "in")
                                          (NP (NP (NN "revenue"))
                                              (PP (IN "in")
                                                  (NP (DT "this"))))))))))
                      (PUNCTUATION COMMA)
                      (NP-TMP (PRP-POSS "its") (JJ "10th")
                              (NN "year")))))))
   (TRACE 8))

(S (NP-SBJ (PRP "it"))
   (VP (MD "can")
       (VP (VBP "open")
           (NP (DT "another") (NN "operation"))
           (PP (ADVP (RB "elsewhere"))
               (IN "in")
               (NP (NNP "Philadelphia")))))
   (TRACE 9))

(DR (EVENT-ENABLES (TRACE 8) (TRACE 9)))
```

Next, TEXTCAP adds grammatical features at the NP level to allow for eventual anaphora resolution. Given a simplified version of sentence 5 above, “Greensgrow sells vegetables.”:

```
(S (NP-SBJ (NNP "Greensgrow"))
   (VP (VBZ "sells")
       (NP (NNS "vegetables"))))
```

One ad-hoc rule matches to the unmodified plural noun and marks it as being a generic class rather than an instance and stems the lexical item. Another rule notes that the subject is a proper noun that is not in its stoplist of person names. As it is not the object of a preposition, it is marked as a company name (via the WordNet sense). Additional senses are assigned if, for instance, only one sense is possible.

```
(S (NP-SBJ (NNP "Greensgrow") (TYPE COMPANY#n1)
           (GENDER NEUTRAL))
   (VP (VBZ "sells")
       (NP (NN "vegetable") (NUMBER PLURAL)
           (GENERIC YES) (GENDER NEUTRAL))))
```

Next, we map grammatical subjects and objects to logical ones, undoing passivization, etc. Then we mark verb type and semantic roles by matching selectional restrictions (currently based on ad-hoc rules) between the verb and its principal arguments. Modifiers of an NP are processed as semantic triples dependent on that NP’s instance, and similarly for verbal modifiers.

```
<relation> = (NP-SBJ (VP (VBZ "sells") NP)
                     (TRACE 5))
           --> (<agent> SELL#v? <patient>)
<agent>    = ((NNP "Greensgrow") (TYPE COMPANY#n1)
           --> COMPANY#n1(name="Greensgrow",
                          gender=neutral)
<patient>  = (NP (NNS "vegetable") ...)
           --> VEGETABLE#n?(generic=yes,
                            gender=neutral,
                            number=plural)
```

Anaphora resolution rules search NPs and their feature lists in reverse to exclude impossible coreferences; TextCap currently uses the first acceptable NP as its coreferent. Next, word sense disambiguation is applied. Because we use WordNet senses as an underlying foundation, we can pass a bag of nearby senses using existing published WSD algorithms, although we are currently testing the degree of performance improvement between simple baselines and custom algorithms. After WSD, we give instance names to each type/sense and drop information on generic entities.

```
<relation> = (<agent> SELL#v1 <patient>), trace=5
<agent>    = COMPANY#n1(name="Greensgrow",
                        inst=COMPANY549)
<patient>  = VEGETABLE#n1
```

Next, we build a list of coindexed semantic triples directly from the above representation. If no sentence-level traces or modifiers are dependent on the verbal relation, it is treated as a generic instance.

```
(COMPANY549 INSTANCE-OF COMPANY#n1)
(COMPANY549 NAME "Greensgrow")
(COMPANY549 SELL#v1 VEGETABLE#n1)
```

After repeating this process for each standard sentence-level parse, triples representing discourse relations are then included for each dependency, for instance:

```
(DR (EVENT-ENABLES (TRACE 8) (TRACE 9)))

(FARM381 MAKING287 PROFIT#n1)
(FARM381 OPENING286 OPERATION#n2)
(MAKING287 EVENT-ENABLES OPENING286)
```

6 Performance on the Shared Task

Overall, TextCap performed well for its intended purpose, but many limitations were encountered on unseen texts, as expected. Principally, word sense disambiguation and pronoun resolution initially caused significant problems in terms of robustness and the capabilities of these text processing steps were significantly downgraded in order that TextCap could run to completion on all seven sets of unseen texts. Thus WSD was run only for WordNet noun senses and pronoun resolution was not run across sentence boundaries within each set. Additionally, the discourse parser lacked rules to correctly convert sentences #1 and #4 in set #5, so the input sentences were manually split in that case.

However, TextCap was able to do a good job at producing semantic triples for every text, and the number of triples was proportional to the length of each sentence, as expected. The use of existing lexical tools and resources allows for more time to be spent on adding and correcting semantic mappings. Some necessary lexical tools are either not available or still limited in terms of accuracy, and some resources do not exist, for instance, there is no good ontological inventory of prepositions and how they should be mapped semantically. In general, overall accuracy (as measured by human inspection) was much better on shorter sentences.

The system performed poorly in some areas such as interpreting questions and quotations involving multiple sentences. Additionally, the structure of many of the triples that TextCap produced were highly reflective on the original syntactic parses — it is not clear, for instance, that they would enable a question answering system to locate correct answers reliably. However, overall, we believe that post-editing of triples with TextCap would provide a significant time speedup compared to manual knowledge engineering alone, and we are looking at methods of showing this empirically.

The following data represent the performance of TextCap on the 2008 STEP shared task. Sentences were processed in an average time of 4 seconds each.

`Set #1`

[1] "An object is thrown with a horizontal speed of 20 m/s from a cliff that is 125 m high."

Notes: (a) the parser interpreted "m/s" as a plural noun; (b) `source` is a very vague relation; (c) `cliff` was correctly recognized as a relative clause subject.

```
((OBJECT001 INSTANCE-OF OBJECT#1)
 (SPEED001 INSTANCE-OF SPEED#1)
 (M/001 INSTANCE-OF M/#0)
 (CLIFF001 INSTANCE-OF CLIFF#1)
 (NUMBER10 INSTANCE-OF NUMBER)
 (UNKNOWN-AGENT THROWING OBJECT001)
 (MANNER-WITH THROWING SPEED001)
 (SPEED001 RANGE-OF M/001)
 (M/001 SOURCE CLIFF001)
 (M/001 NUMERIC-QUANTITY 20)
 (SPEED001 CHARACTERISTICS HORIZONTAL)
 (TIME-PERIOD THROWING PRESENT)
 (NUMBER10 HAS-VALUE 125)
 (CLIFF001 BEING NUMBER10))
```

[2] "The object falls for the height of the cliff."

Notes: (a) `for` was incorrectly intepreted as purpose ("object" would be animate); (b) `of` yielded the wrong relation; (c) it's unclear what should be the 3rd element of the triple for `falling`.

```
((OBJECT001 INSTANCE-OF OBJECT#1)
 (HEIGHT001 INSTANCE-OF HEIGHT#1)
 (CLIFF001 INSTANCE-OF CLIFF#1)
 (OBJECT001 FALLING INTRANSITIVE-ARGUMENT)
 (FALLING PURPOSE-FOR HEIGHT001)
 (HEIGHT001 RANGE-OF CLIFF001))
```

[3] "If air resistance is negligible, how long does it take the object to fall to the ground?"
Notes: This sentence was not processed satisfactorily due to no rules to detect questions of the form "how [adjp]".

[4] "What is the duration of the fall?"
Notes: This sentence was processed satisfactorily.

Set #2

[1] "Cervical cancer is caused by a virus."
Notes: (a) probably better to map `cervical` to `cervix` to allow for semantic processing in, e.g., a question answering system.

```
((VIRUS001 INSTANCE-OF VIRUS#1)
 (CANCER001 INSTANCE-OF CANCER#1)
 (VIRUS001 CAUSING CANCER001)
 (CANCER001 CHARACTERISTICS CERVICAL))
```

[2 "That has been known for some time and it has led to a vaccine that seems to prevent it."
Notes: (a) the system has more trouble mapping situational referents, but it did correctly notice one was present; (b) need more mappings for `for` besides purpose; (c) need to map from grammatical tense to relational tense.

```
((SITUATION001 INSTANCE-OF UNKNOWN-REFERENT)
 (TIME001 INSTANCE-OF TIME#1)
 (VACCINE001 INSTANCE-OF VACCINE#1)
 (SITUATION001 KNOWING INTRANSITIVE-ARGUMENT)
 (KNOWING PURPOSE-FOR TIME001)
 (TIME001 QUANTIFIER-VALUE SOME)
 (TIME-PERIOD KNOWING PAST-HABITUAL-ACTION)
 (SINGLE-NEUTER-REFERENT LEADING INTRANSITIVE-ARGUMENT)
 (DESTINATION LEADING VACCINE001)
 (TIME-PERIOD LEADING PRESENT-PERFECT)
 (VACCINE001 PREVENTING001 SINGLE-NEUTER-REFERENT)
 (VACCINE001 SEEMING PREVENTING001))
```

[3] "Researchers have been looking for other cancers that may be caused by viruses."
Notes: (a) didn't map `looking` and `for` as a single verbal relation; (b) the treatment of quantifiers is too simplistic (`other`).

```
((RESEARCHER001 INSTANCE-OF RESEARCHER#1)
 (CANCER001 INSTANCE-OF CANCER#1)
 (VIRUS001 INSTANCE-OF VIRUS#1)
 (RESEARCHER001 LOOKING INTRANSITIVE-ARGUMENT)
 (RESEARCHER001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (LOOKING PURPOSE-FOR CANCER001)
 (CANCER001 CHARACTERISTICS OTHER-ADJ)
 (CANCER001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (TIME-PERIOD LOOKING PAST-HABITUAL-ACTION)
 (VIRUS001 CAUSING CANCER001)
 (VIRUS001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (MODALITY CAUSING MODAL-MAY))
```

Set #3

We skip this set of sentences as TEXTCAP seemed to perform very well on set #3 excepting the pronouns in sentence #5.

Set #4

[1] "The first school for the training of leader dogs in the country is going to be created in Mortagua and will train 22 leader dogs per year."
Notes: (a) `Mortagua` wasn't treated as a city name; (b) incorrect treatment of complex passive verb phrases ("going" is not a main verb); the same semantic object `school` is correctly noted as being involved in both phrases.

```
((SCHOOL001 INSTANCE-OF SCHOOL#1)
 (MORTAGUA001 INSTANCE-OF PERSON#1)
 (MORTAGUA001 ACTOR-NAME "Mortagua")
 (MORTAGUA001 ACTOR-GENDER NEUTER)
 (DOG001 INSTANCE-OF DOG#1)
 (YEAR001 INSTANCE-OF YEAR#1)
 (SCHOOL001 GOING INTRANSITIVE-ARGUMENT)
 (SCHOOL001 CHARACTERISTICS FIRST)
 (GOING LOCATION-IN MORTAGUA001)
 (TIME-PERIOD GOING PRESENT-PROGRESSIVE)
 (SCHOOL001 TRAINING DOG001)
 (DOG001 PER YEAR001)
 (DOG001 NAMED-TYPE LEADER#1)
 (DOG001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (TIME-PERIOD TRAINING FUTURE))
```

[2] "In Mortagua, Joao Pedro Fonseca and Marta Gomes coordinate the project that seven people develop in this school."
Notes: This sentence was processed satisfactorily.

[3] "They visited several similar places in England and in France, and two future trainers are already doing internship in one of the French Schools."
Notes: (a) not a good quantifier representation for `several`; (b) any proper NP is being interpreted as a person.

```
((PLACE001 INSTANCE-OF PLACE#1)
 (TRAINER001 INSTANCE-OF TRAINER#1)
 (INTERNSHIP001 INSTANCE-OF INTERNSHIP#1)
 (NUMBER11 INSTANCE-OF NUMBER)
 (FRENCH-SCHOOLS001 INSTANCE-OF PERSON#1)
 (FRENCH-SCHOOLS001 ACTOR-NAME "French Schools")
 (FRENCH-SCHOOLS001 ACTOR-GENDER NEUTER)
 (PLURAL-THIRD-PERSON-REFERENT VISITING PLACE001)
 (PLACE001 CHARACTERISTICS SEVERAL)
 (PLACE001 CHARACTERISTICS SIMILAR)
 (PLACE001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (TIME-PERIOD VISITING PAST)
 (TRAINER001 DOING INTERNSHIP001)
 (TRAINER001 WRITTEN-NUMERIC-QUANTITY 2)
 (TRAINER001 CHARACTERISTICS FUTURE)
 (TRAINER001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (DURATION DOING ALREADY)
 (DOING LOCATION-IN NUMBER11)
 (NUMBER11 RANGE-OF FRENCH-SCHOOLS001)
 (TIME-PERIOD DOING PRESENT-PROGRESSIVE))
```

[4] “The communitarian funding ensures the operation of the school until 1999.”
Notes: This sentence was relatively uninteresting.

[5] “We would like our school to work similarly to the French ones, which live from donations, from the merchandising and even from the raffles that children sell in school.”
Notes: This sentence was not processed satisfactorily due to missing discourse parsing rules.

Set #5

[1] “As the 3 guns of Turret 2 were being loaded, a crewman who was operating the center gun yelled into the phone, ‘I have a problem here. I am not ready yet.’ ”
Notes: (a) this sentence was manually split before the quotation; (b) another proper NP interpreted as a person; (c) the system in general works well with quotations, but not when they are composed of multiple sentences.

```
((GUN001 INSTANCE-OF GUN#1)
 (TURRET-2001 INSTANCE-OF PERSON#1)
 (TURRET-2001 ACTOR-NAME "Turret 2")
 (TURRET-2001 ACTOR-GENDER NEUTER)
 (CREWMAN001 INSTANCE-OF CREWMAN#1)
 (CENTER-GUN001 INSTANCE-OF CENTER-GUN#0)
 (PROBLEM001 INSTANCE-OF PROBLEM#1)
 (UNKNOWN-AGENT LOADING GUN001)
 (GUN001 RANGE-OF TURRET-2001)
 (GUN001 NUMERIC-QUANTITY 3)
 (TIME-PERIOD LOADING PAST-PROGRESSIVE)
 (CREWMAN001 OPERATING CENTER-GUN001)
 (TIME-PERIOD OPERATING PAST-PROGRESSIVE)
 (PROBLEM001 BEING READY)
 (DURATION BEING YET)
 (POLARITY BEING NEGATIVE))
```

[2] “Then the propellant exploded.”
Notes: This sentence was processed satisfactorily.

[3] “When the gun crew was killed they were crouching unnaturally, which suggested that they knew that an explosion would happen.”
Notes: This sentence presented more syntactic than semantic issues.

[4] “The propellant that was used was made from nitrocellulose chunks that were produced during World War II and were repackaged in 1987 in bags that were made in 1945.”
Notes:

[5] “Initially it was suspected that this storage might have reduced the powder’s stability.”

Notes: (a) the possessive noun `powder` was incorrectly marked as a person; (b) the `time` and `modality` markers are a bit vague.

```
((STORAGE001 INSTANCE-OF STORAGE#1)
 (STABILITY001 INSTANCE-OF STABILITY#1)
 (POWDER001 INSTANCE-OF PERSON#1)
 (POWDER001 ACTOR-NAME "powder")
 (POWDER001 ACTOR-GENDER NEUTER)
 (UNKNOWN-AGENT SUSPECTING REDUCING)
 (STORAGE001 REDUCING STABILITY001)
 (TIME SUSPECTING INITIALLY)
 (TIME-PERIOD REDUCING PRESENT-PERFECT)
 (MODALITY REDUCING MODAL-MIGHT)
 (TIME-PERIOD SUSPECTING PAST))
```

`Set #6`

Data in this set was used to test TEXTCAP and so is not analyzed here.

`Set #7`

[1] “Modern development of wind-energy technology and applications was well underway by the 1930s, when an estimated 600,000 windmills supplied rural areas with electricity and water-pumping services.”

Notes: (a) couldn’t convert `1930s` to a date range; (b) `underway` was treated as a verb by the parser; (c) more problems mapping prepositional relations.

```
((DEVELOPMENT001 INSTANCE-OF DEVELOPMENT#1)
 (TECHNOLOGY001 INSTANCE-OF TECHNOLOGY#1)
 (APPLICATION001 INSTANCE-OF APPLICATION#1)
 (NUMBER24 INSTANCE-OF NUMBER)
 (NUMBER24 HAS-VALUE "1930")
 (WINDMILL001 INSTANCE-OF WINDMILL#1)
 (AREA001 INSTANCE-OF AREA#1)
 (ELECTRICITY001 INSTANCE-OF ELECTRICITY#1)
 (SERVICE001 INSTANCE-OF SERVICE#1)
 (UNKNOWN-AGENT UNDERWAY DEVELOPMENT001)
 (DEVELOPMENT001 RANGE-OF TECHNOLOGY001)
 (DEVELOPMENT001 RANGE-OF APPLICATION001)
 (TECHNOLOGY001 CHARACTERISTICS WIND-ENERGY)
 (APPLICATION001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (DEVELOPMENT001 CHARACTERISTICS MODERN)
 (DURATION UNDERWAY WELL)
 (TIME-BY UNDERWAY NUMBER24)
 (NUMBER24 NUMBER-OF-UNITS MORE-THAN-ONE)
 (TIME-PERIOD UNDERWAY PAST)
 (WINDMILL001 SUPPLYING AREA001)
 (WINDMILL001 NUMERIC-QUANTITY 600000)
 (AREA001 HAVE-WITH ELECTRICITY001)
 (AREA001 HAVE-WITH SERVICE001)
 (SERVICE001 NAMED-TYPE WATER-PUMPING#0)
 (SERVICE001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (AREA001 CHARACTERISTICS RURAL)
 (AREA001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (TIME-PERIOD SUPPLYING PAST))
```

[2] “Once broad-scale electricity distribution spread to farms and country towns, use of wind energy in the United States started to subside, but it picked up again after the U.S. oil shortage in the early 1970s.”
Notes: Notes: This sentence was processed satisfactorily, but only when manually split due to missing discourse parsing rules.

[3] “Over the past 30 years, research and development has fluctuated with federal government interest and tax incentives.”
Notes: This sentence was processed satisfactorily.

[4] “In the mid-’80s, wind turbines had a typical maximum power rating of 150 kW.”
Notes: This sentence had problems understanding the phrase “mid-’80s”, perhaps as a result of the off-the-shelf parser being very generic.

[5] “In 2006, commercial, utility-scale turbines are commonly rated at over 1 MW and are available in up to 4 MW capacity.”
Notes: (a) the fact that someone rates turbines isn’t the same as turbines carrying a rating; `commonly` wasn’t interpreted correctly; (c) the last phrase after `available` wasn’t mapped to anything.

```
((TURBINE001 INSTANCE-OF TURBINE#1)
 (DATE26 INSTANCE-OF DATE)
 (DATE26 HAS-YEAR 2006)
 (UNKNOWN-AGENT RATING TURBINE001)
 (TURBINE001 CHARACTERISTICS COMMERCIAL)
 (TURBINE001 CHARACTERISTICS UTILITY-SCALE)
 (TURBINE001 NUMBER-OF-UNITS MORE-THAN-ONE)
 (FREQUENCY RATING COMMONLY)
 (TIME-IN RATING DATE26)
 (TURBINE001 BEING AVAILABLE)
 (TURBINE001 NUMBER-OF-UNITS MORE-THAN-ONE))
```

7 Conclusions

We introduced TEXTCAP, a semantic parser which uses a combination of off-the-shelf NLP technology and ad-hoc rules to produce semantic triples corresponding to the explicit semantic content in unrestricted text. We ran TEXTCAP on 7 sets of short text in the STEP 2008 Shared Task, and the system successfully generated triples for almost all inputs and provided, as we expected, a set of triples that while not fully correct, could be post-edited for accuracy and which should provide a significant speed up over completely manual production of semantic triples from text. On average, TEXTCAP processed a sentence from the corpus in about 4 seconds.

While TEXTCAP only captures explicit knowledge (but not commonsense knowledge, unmentioned knowledge, implicit relationships, etc.) it can save knowledge engineers time by providing reasonably accurate semantic representations of domain text. In future work we plan on improving methods of knowledge integration (e.g., ontology population), testing within real-world applications such as question answering systems, and empirically evaluating the time and accuracy for producing semantic triples via various methods.

References

Barker, K., B. Agashe, S. Chaw, J. Fan, N. Friedland, M. Glass, J. Hobbs, E. Hovy, D. Israel, D. S. Kim, R. Mulkar-Mehta, S. Patwardhan, B. Porter, D. Tecuci, and P. Yeh (2007, July). Learning by reading: A prototype system, performance baseline and lessons learned. In *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI)*, Vancouver, Canada.

Blythe, J., J. Kim, S. Ramachandran, and Y. Gil (2001). An integrated environment for knowledge acquisition. In *Proceedings of the 2001 International Conference on Intelligent User Interfaces*, Santa Fe, NM, USA.

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.

Brachman, R. J. and J. G. Schmolze (1985, April). An overview of the KL-ONE knowledge representation system. *Cognitive Science 9*(2), 171–216.

Callaway, C., E. Not, A. Novello, C. Rocchi, O. Stock, and M. Zancanaro (2005, June). Automatic cinematography and multilingual NLG for generating video documentaries. *Artificial Intelligence 165*(1), 57–89.

Carenini, G., R. T. Ng, and E. Zwart (2005). Extracting knowledge from evaluative text. In *K-CAP '05: Proceedings of the 3rd International Conference on Knowledge Capture*, Banff, Canada, pp. 11–18.

Charniak, E. (2000, April). A maximum-entropy-inspired parser. In *Proceedings of the 2000 NAACL*, Seattle, WA.

Clark, P. and B. Porter (1998). KM – the knowledge machine: Users manual. Technical report, AI Lab, University of Texas at Austin.

Clark, P., J. Thompson, K. Barker, B. Porter, V. Chaudhri, A. Rodriguez, J. Thomr, Y. Gil, and P. Hayes (2001, October). Knowledge entry as the graphical assembly of components: The SHAKEN system. In *Proceedings of the First International Conference on Knowledge Capture (KCAP)*, Victoria BC, Canada.

Collins, M. (1999). *Head-driven Statistical Models for Natural Language Parsing*. Ph. D. thesis, University of Pennsylvania.

Fellbaum, C. (1998). *WordNet: An electronic lexical database*. The MIT Press.

Gildea, D. and D. Jurafsky (2002). Automatic labeling of semantic roles. *Computational Linguistics 28*(3), 245–288.

Gliozzo, A., C. Giuliano, and C. Strapparava (2005, June). Domain kernels for word sense disambiguation. In *Proceedings of the 43th Annual Meeting of the Association for Computational Linguistics*, Ann Arbor, MI, pp. 403–410.

Lester, J. C. and B. W. Porter (1997). Developing and empirically evaluating robust explanation generators: The KNIGHT experiments. *Computational Linguistics 23*(1), 65–101.

Mann, W. C. and S. A. Thompson (1987, June). Rhetorical structure theory: A theory of text organization. Technical Report ISI/RS-87-190, USC/Information Sciences Institute, Marina del Rey, CA.

Marcus, M., B. Santorini, and M. Marcinkiewicz (1993). Building a large annotated corpus of English: The PennTreeBank. *Computational Linguistics 19*(2), 313–330.

Poesio, M. and M. A. Kabadjov (2004, May). A general-purpose, off-the-shelf system for anaphora resolution. In *Proceedings of the Language Resources and Evaluation Conference*, Lisbon, Portugal.

Reiter, E., S. Sripada, and R. Robertson (2003). Acquiring correct knowledge for natural language generation. *Journal of Artificial Intelligence Research 18*, 491–516.

Soricut, R. and D. Marcu (2003, May). Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of HLT-NAACL*, Edmonton, Alberta.

Stock, O., M. Zancanaro, P. Busetta, C. Callaway, A. Krueger, M. Kruppa, T. Kuflik, E. Not, and C. Rocchi (2007). Adaptive, intelligent presentation of information for the museum visitor in peach. *User Modeling and User Adapted Interaction 17*, 257–304.

Yeh, P., B. Porter, and K. Barker (2006, July). A unified knowledge based approach for sense disambiguation and semantic role labeling. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, Boston, MA.