# Abduction for Discourse Understanding

This paper addresses the challenge of learning information by reading natural language text. The major aim is to map natural language input into logical expressions anchored upon concise and specific theories underlying the domains, in such a way that a reasoning engine can be used to answer questions about the input. We define a 3-step procedure, including parsing and abduction, and explore different implementations for the steps. Experiments were conducted in the "Engine" domain, but the versatility of the approach suggests that extension to other domains is possible when the underlying theories are suitably specified.

## Background

The word done in this paper, is part of the effort towards the DARPA funded - Mobius Project. The projct aims at a Learning by Reading approach to text, having a Natural Language Engine and a Knowledge Reasoning engine. The Knowledge Reasoning Engine received information from the Natural Language Engine and created new models and concepts from each sentence that it processed. The domain for this effort was on texts related to Engines.

## Language Processing

The entire Language Processing component is called the NL component. NL consists of 3 major components: Parser, Logical form converter and Abduction Engine. The text is parsed, a shallow logical form is generated, and abductive inference is used to convert the content of the text to the language required by the KM [Clark et al., 2003] system for matching its models of devices. We here refer to this entire process as NL.

### Parsing

The first step of parsing was performed by the Charniak parser [Charniak, 2000]. Below is a sample parse for the sentence

"An engine burns gasoline."

(S1 (S (NP (DT An) (NN engine)) (VP (VBZ burns) (NP (NN gasoline))) (. .)))

### Shallow logical form

The parse trees are translated into a shallow logical form as defined by Hobbs in [Hobbs 85,98]. Logical form created for ``An engine burns gasoline." is

an'(e5,x0,e4) & engine-nn'(e4,x0) & burn-vb'(e0,x0,x1) & Present-tense'(e1,e0) & gasoline-nn'(e6,x1)

Here x0 and x1 represent the entity engine and gasoline, and e4 represents the eventuality of existence of the burning event.

LFToolkit [Rathod and Hobbs 2005] is used to convert parse tree outputs to a shallow logical form. LFToolkit works by generating logical form fragments corresponding to lexical items in the sentence and using syntactic composition relations to identify variables among logical form fragments. In this step, certain additional logical representation forms may be introduced, such as, for plural nouns, sets with variables that represent the individuals as well as the set implicit in the plural noun itself.

### Transformations and Mapping

The mapping from this logical form fragments to the triple representation required by KM is performed by Mini-Tacitus [Hobbs 93]. This step involves fragment combination, integration, some reformulation, as well as, the introduction of additional supporting or inferentially derived material. It is performed by backchaining and making assumptions where necessary, using a knowledge base of axioms. In the above example the output produced after abduction is:

an'(e5,x0,e4) & engine-nn'(e4,x0) & device-ot'(E21,x0) & burn-vb'(e0) & combustion-vt'(E13,e0,x0,x1) & Present-tense'(e1,e0) & gasoline-nn'(e6,x1) & power-ot'(E24,x1) & fuel-ot'(E3,x1) & Liquid-Substance-ot'(E4,x1)

### Harvesting Axioms

The development for axioms for Mini-TACITUS required classifying of nouns and verbs into general categories, for limiting the total interpretation space, and avoiding explosion of interpretation. Also, this enabled categorizing nouns and verbs into selective classes and provided extra information about the nouns and verbs in a specific class.

The axioms for Mini-TACITUS were developed in a semi-automatic manner, using the statistics of word relationships as described by [Pantel and Lin 2000]. The corpus consisted of 1400 web scraped texts, related to engines. These texts were parsed using Minipar parser [Lin 98]. Using the parse, the verb and noun modifiers/relations with their counts were extracted and their mutual information was computed [Pantel and Lin 2000]. From this available pool of information, all the verbs were picked and their possible subject's and object's were short listed based on the features and Mutual Information provided. similar propositions to produce a final linked interpretation of the originally unlinked piece of text.

### Results

The addition of Axioms to the abduction engine, improved the total model and concept creation by KR. There was an evident reduction in the Interpretation space for Mini-TACITUS, as interpretation was mainly at the class level rather than lexical level. Effective funneling was performed where lexical tokens were handled collectively as a class rather than individual lexical units. The total verb linkages in text increased and there missing linkages due to broken parses were also patched. Using abduction has shown a very promising outcome, and we are not investigating on automating the entire axiom development process.

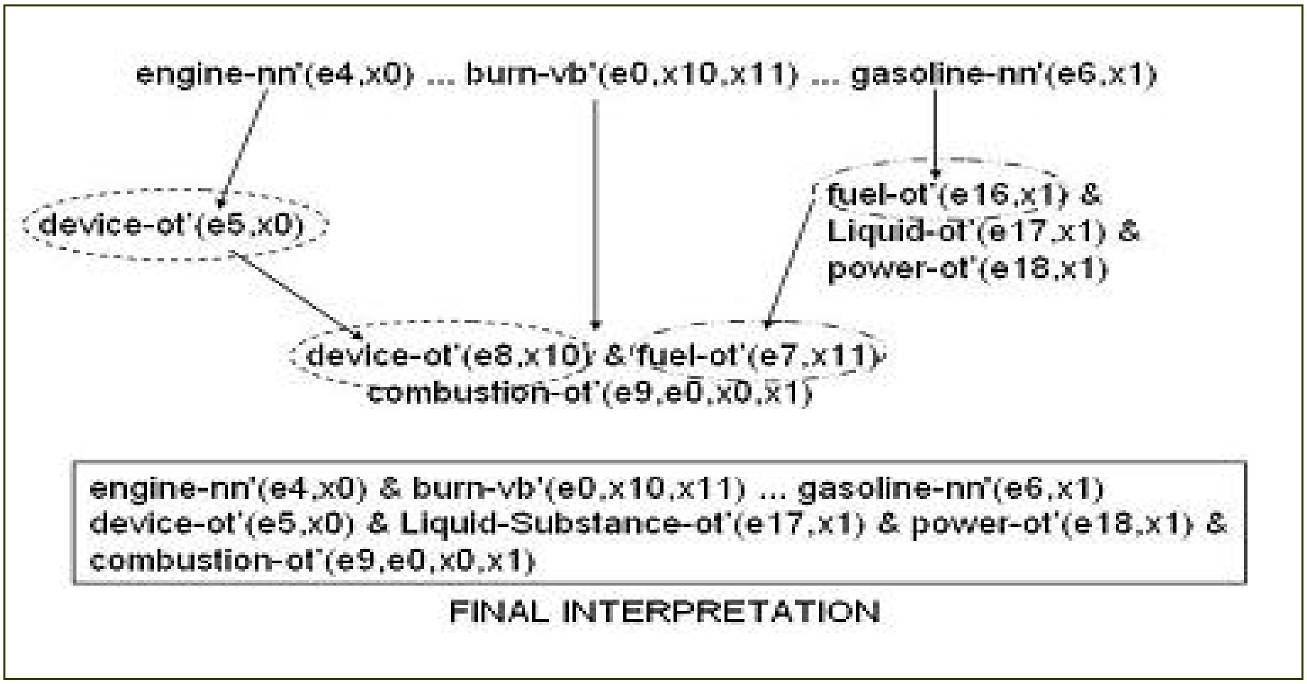


**Figure 1: Example of Abduction process**

Figure 1 is an example of how Abduction produces extra information about the domain nouns and verbs. Here we can see that x0 is an engine, which is a type of device (indicated by the ot suffix). e0 is the burning event which is a subtype of the verb class `combustion' (indicated by the suffix vt. There is unification of

### References


**[Charniak, 2000]** Eugene Charniak. *A maximum-entropy inspired parser*. In Proceedings of the first conference on North American chapter of the Association for Computational Linguistics, pages

132.139, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

**[Clark et al., 2003]** Peter Clark, Phil Harrison, and John Thompson. *A knowledge-driven approach to text meaning processing*. In Proceedings of the HLT-NAACL 2003 workshop on Text meaning - Volume 9, pages 1 . 6, 2003.

**[Hobbs et al., 1993]** Jerry Hobbs, Mark Stickel, Douglas
Appelt, and Paul Martin. *Interpretation as abduction*. In Artificial Intelligence Vol. 63, Nos. 1-2, pp. 69-142, 1993.

**[Hobbs, 1985]** Jerry R. Hobbs. *Ontological promiscuity*. In Proceedings, 23rd Annual Meeting of the Association for Computational Linguistics, pages 61.69, 1985.

**[Hobbs, 1998]** Jerry Hobbs. *The logical notation: Ontological promiscuity*. In Discourse and Inference: Magnum Opus in Progress, 1998.

**[Rathod and Hobbs, 2005]** Nishit Rathod and Jerry Hobbs. *LFTookit*. In http://www. isi.edu/.nrathod /wne /LFToolkit /index.html, 2005.

**[Pantel and Lin. 2000]** *Word-for-Word Glossing with Contextually Similar Words*. In Proceedings of Applied Natural Language Processing / North American Association for Computational Linguistics (ANLP/NAACL-00). pp. 78-85. Seattle, WA

**[D. Lin.]** *Dependency-based evaluation of minipar*. In Proceedings of Workshop on the Evaluation of Parsing Systems, First International Conference on Language Resources and Evaluation., 1998.