

Augmenting WordNet for Deep Understanding of Text

Peter Clark¹

Christiane Fellbaum²

Jerry R. Hobbs³

Phil Harrison¹

William R. Murray¹

John Thompson¹

¹The Boeing Company (USA)

²Princeton University (USA)

³University of Southern California (USA)

email: `peter.e.clark@boeing.com`

Abstract

One of the big challenges in understanding text, i.e., constructing an overall coherent representation of the text, is that much information needed in that representation is unstated (implicit). Thus, in order to “fill in the gaps” and create an overall representation, language processing systems need a large amount of world knowledge, and creating those knowledge resources remains a fundamental challenge. In our current work, we are seeking to augment WordNet as a knowledge resource for language understanding in several ways: adding in formal versions of its word sense definitions (glosses); classifying the morphosemantic links between nouns and verbs; encoding a small number of “core theories” about WordNet’s most commonly used terms; and adding in simple representations of scripts. Although this is still work in progress, we describe our experiences so far with what we hope will be a significantly improved resource for the deep understanding of language.

1 Introduction

Much information that text is intended to convey is not explicitly stated. Rather, the reader constructs a mental model of the scene described by the text, including many “obvious” features that were not explicitly mentioned. By one estimate, the ratio of explicit to implicit facts is 1:8 (Graesser, 1981), making the task of understanding text, i.e., constructing a coherent representation of the scene that the author intended to convey, very difficult, even given the generally reasonable quality of syntactic interpretation that today’s systems produce. For example, given the sentence:

A soldier was killed in a gun battle.

a reader will infer that (probably):

The soldier was shot; The soldier died; There was a fight; etc.

even though none of these facts are explicitly stated. A person is able to draw these plausible conclusions because of the large amounts of world knowledge he/she has, and his/her ability to use them to construct an overall mental model of the scene being described.

A key requirement for this task is access to a large body of world knowledge. However, machines are currently poorly equipped in this regard. Although a few knowledge encoding projects are underway, e.g., Cyc (Lenat and Guha, 1989), developing such resources continues to be a major challenge, and any contribution to this task has significant potential benefit. WordNet (Miller, 1995; Fellbaum, 1998) presents an unique avenue for making inroads into this problem: It already has broad coverage, multiple lexicosemantic connections, and significant knowledge encoded (albeit informally) in its glosses. It can thus be viewed as on the way to becoming an extensively leveragable, “lightweight” knowledge base for reasoning. In fact, WordNet already plays a central role in many question-answering systems e.g., 21 of the 26 teams in the recent PASCAL RTE3 challenge used WordNet (Giampiccolo et al., 2007), and most other large-scale resources already include mappings to it and thus can leverage it easily. In our work we are developing several augmentations to WordNet to improve its utility further, and we report here on our experiences to date.

Although we are performing experiments with recognizing textual entailment (RTE) (determining whether a hypothesis sentence *H* follows from some text *T*), it is important to note that RTE is not our end-goal. Many existing RTE systems, e.g., (Adams et al., 2007; Chambers et al., 2007) largely work by statistically scoring the match between *T* and *H*, but this to an extent sidesteps “deep” language understanding, namely building a coherent, internal representation of the overall scenario the input text was intended to convey. RTE is one way of measuring success in this endeavor, but it is also possible to do moderately well in RTE without the system even attempting to “understand” the scenario the text is describing. It is yet to be seen whether very high performance in RTE can be obtained without some kind of deep language understanding of the entire scene that a text conveys.

We are testing our work with BLUE, Boeing’s Language Understanding Engine, which we first describe. We then present the WordNet augmentations that we are developing, and our experience with these as well as with the DIRT paraphrase database.

The contribution of this paper is some preliminary insight into avenues and challenges for creating and leveraging more world knowledge, in the context of WordNet, for deeper language understanding.

2 Text Interpretation and Subsumption

2.1 Text Interpretation

For text interpretation we are using BLUE, Boeing’s Language Understanding Engine (Clark and Harrison, 2008), comprising a parser, logical form (LF) generator, and final logic 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, loosely based on Schubert and Hwang (1993). The LF is a simplified and normalized tree structure with logic-type elements, generated by rules parallel to the grammar rules, that contains variables for noun phrases and additional expressions for other sentence constituents. 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. A simple example of an LF is shown below (items starting with underscores `_` denote variables):

```
;;; LF for "A soldier was killed in a gun battle."
(DECL ((VAR _X1 "a" "soldier")
       (VAR _X2 "a" "battle" (NN "gun" "battle"))
       (S (PAST) NIL "kill" _X1 (PP "in" _X2))))
```

The LF is then used to generate ground logical assertions of the form $r(x,y)$, containing Skolem instances, by applying a set of syntactic rewrite rules recursively to it. Verbs are reified as individuals, Davidsonian-style. An example of the output is:

```
;;; logic for "A soldier was killed in a gun battle."
object(kill01,soldier01)
in(kill01,battle01)
modifier(battle01,gun01)
```

plus predicates associating each Skolem with its corresponding input word. At this stage of processing, the predicates are syntactic relations ($\text{subject}(x,y)$, $\text{object}(x,y)$, $\text{modifier}(x,y)$, and all the prepositions, e.g., $\text{in}(x,y)$). Definite coreference is computed by a special module which uses the (logic for the) referring noun phrase as a query on the database of assertions. Another module performs special structural transformations, e.g., when a noun or verb should map to a predicate rather than an individual. Two additional modules perform (currently naive) word sense disambiguation (WSD) and semantic role labelling (SRL), described further in Clark and Harrison (2008). However, for our RTE experiments we have found it more effective to leave senses and roles underspecified, effectively considering all valid senses and roles (for the given lexical features) during reasoning until instantiated by the rules that apply.

2.2 Subsumption

A basic operation for reasoning is determining if one set of clauses subsumes (is more general than, is thus implied by) another, e.g., (the logic for) “A person likes a person” subsumes “A man loves a woman”. This basic operation is used both to determine if an axiom applies, and in RTE to determine if a text H subsumes (is implied by) a text T or its axiom-expanded elaboration. A set S1 of clauses subsumes another S2 if each clause in S1 subsumes some (different) member of S2. A clause C1 subsumes another C2 if both (for binary predicates) of C1’s arguments subsume the corresponding arguments in C2, and C1 and C2’s predicates “match”. An argument A1 subsumes another A2 if some word sense for A1’s associated word is equal or more general (a hypernym of) some word sense of A2’s associated word (thus effectively considering all possible word senses for A1 and A2)¹. We also consider adjectives related by WordNet’s “similar” link, e.g., “clean” and “pristine”, to be equal. Two syntactic predicates “match” (i.e., are considered to denote the same semantic relation) according to the following rules:

1. both are the same;
2. either is the predicate “of” or “modifier”;
3. the predicates “subject” and “by” match (for passives);
4. the two predicates are in a small list of special cases that should match e.g., “on” and “onto”.

These rules for matching syntactic roles are clearly an approximation to matching semantic roles, but have performed better in our experiments than attempting to explicitly assign (with error) semantic roles early on and then matching on those.

In addition, in language, ideas can be expressed using different parts of speech (POS) for the same basic notion, e.g., verb or noun as in “The bomb destroyed the shrine” or “The destruction of the shrine by the bomb” (Gurevich et al., 2006). To handle these cross-POS variants, when finding the word senses of a word (above) our system considers all POS, independent of its POS in the original text. Combined with the above predicate-matching rules, this is a simple and powerful way of aligning expressions using different POSs, e.g.:

- “The bomb destroyed the shrine” and “The destruction of the shrine by the bomb” (but not “The destruction of the bomb by the shrine”) are recognized as equivalent.
- “A person attacks with a bomb” and “There is a bomb attack by a person” are recognized as equivalent.
- “There is a wrecked car”, “The car was wrecked”, and “The car is a wreck” (adjective, verb, and noun forms) are recognized as equivalent.

Although clearly these heuristics can go wrong, they provide a basic mechanism for assessing simple equivalence and subsumption between texts.

¹Clearly this can go wrong, e.g., if the contexts of T and H are different so repeated/matching words have incompatible intended senses, although such discontinuities are unusual in natural text.

2.3 Experimental Test Bed

As an experimental test bed we have developed a publically available RTE-style test suite² of 250 pairs (125 entailed, 125 not entailed). As our goal is deeper semantic processing, the texts are syntactically simpler than the PASCAL RTE sets (at www.pascal-network.org) but semantically challenging to process. We use examples from this test suite (and others) in this paper.

3 Exploiting Lexical & World Knowledge

3.1 Use of WordNet’s Glosses

Translation to Logic WordNet’s word sense definitions (glosses) appear to contain substantial amounts of world knowledge that could help with semantic interpretation of text, and we have been exploring leveraging these by translating them into first-order logic. We have also experimented with Extended WordNet (XWN), a similar database constructed several years ago by Moldovan and Rus (2001).

To do the translation, a different language interpreter, developed by ISI, was used (for historic reasons — BLUE was not available at the time the translations were done, and has not been exercised or extended for definition processing). ISI’s system works as follows: First each gloss is converted into a sentence of the form “word is gloss” and parsed using the Charniak parser. Then the parse tree is then converted into a logical syntax by a system called LFToolkit, developed by Nishit Rathod. In LFToolkit, lexical items are translated into logical fragments involving variables. Finally, as syntactic relations are recognized, variables in the constituents are identified as equal. For example, “John works” is translated into $\text{John}(x1) \ \& \ \text{work}(e,x2) \ \& \ \text{present}(e)$, where e is a working event, and then a rule which recognizes “John” as the subject of “works” sets $x1$ and $x2$ equal to each other. Rules of this sort were developed for a large majority of English syntactic constructions. ISI’s system was then used to translate the modified WordNet glosses into axioms. For example (rewritten from the original eventuality notation):

```
;;; "ambition#n2: A strong drive for success"
ambition(x1) -> a(x1) & strong(x1) & drive(x1) & for(x1,x6) & success(x6)
```

Predicates are assigned word senses using the new-ly released WordNet sense-tagged gloss corpus³. This process was applied to all $\approx 110,000$ glosses, but with particular focus on glosses for the 5,000 “core” (most frequently used) synsets. It resulted in good translations for 59.4% of the 5,000 core glosses, with lower quality for the entire gloss corpus. Where there was a failure, it was generally the result of a bad parse, with constructions for which no LFToolkit rules had been written. In these cases, the constituents are translated into logic, so that no information is lost; what is lost is the equalities between variables that provides the connections between the constituents. For instance, in the “John works” example, we would know that there was someone named John and that somebody works, but we would not know that they were the same person. Altogether 98.1% of the 5,000 core glosses were translated into correct axioms (59.4%) or axioms that had all the propositional content but were

²<http://www.cs.utexas.edu/~pclark/bpi-test-suite/>

³<http://wordnet.princeton.edu/glosstag>

disconnected in this way (38.7%). The remaining 1.9% of these glosses had bizarrely wrong parses due to noun-adjective ambiguities or to complex conjunction ambiguities.

Using the Glosses We have used a combination of these logicalized glosses and those from XWN to infer implicit information from text. Although the quality of the logic is generally poor (for a variety of reasons, in particular that the glosses were never intended for machine processing in the first place), our software was able to infer conclusions that help answer a few entailment problems, for example:

T: Britain puts curbs on immigrant labor from Bulgaria and Romania.
H: Britain restricted workers from Bulgaria.

using the logic for the definition:

```
restrict#v1: "restrict", "restrain": place limits on.
```

plus WordNet's knowledge that: "put" and "place" are synonyms; "curb" and "limit" are synonyms; and a laborer is a worker. In our experiments, the glosses were used to answer 5 of the 250 entailment questions (4 correctly). More commonly, the glosses came "tantalizingly close" to providing the needed knowledge. For example, for:

T: A Union Pacific freight train hit five people.
H: A locomotive was pulling the train.

it seems that the definition:

```
train#n1: "train", "railroad train": public transport provided  
by a line of railway cars coupled together and drawn by a locomotive.
```

is very close to providing the needed knowledge. However, unfortunately it defines a train as "public transport provided by cars pulled by a locomotive" rather than just "cars pulled by a locomotive" (the locomotive pulls the cars, not the train/public-transport), hence the hypothesis H is not concluded. Similarly:

T: The Philharmonic orchestra draws large crowds.
H: Large crowds were drawn to listen to the orchestra.

essentially requires knowledge that crowds (typically) listen to orchestras. WordNet's glosses come very close to providing this, with knowledge that:

```
orchestra = collection of musicians  
musician = someone who plays musical instrument  
music = sound produced by musical instruments  
listen = hear = perceive sound
```

However, the connection that the playing results in sound production is missing, and hence again H cannot be inferred. These experiences with the WordNet glosses were very common. In summary, our experience is the WordNet glosses provided some value, being used 5 times (4 correctly) on the 250 examples in our test suite,

with the short, simple definitions (e.g., bleed = lose blood) being the most reliable. The low quality of the logic was a problem (definitional text is notoriously difficult to interpret automatically (Ide and Veronis, 1993)), although often the knowledge came close. Finally, 110,000 rules (approx. one per gloss) is actually quite a small number; typically only 10's of rules fired per sentence, rarely containing the implications we were looking for.

3.2 Typed Morphosemantic Links

WordNet contains approximately 21,000 links connecting derivationally related verb and noun senses, e.g., employ#v2-employee#n1; employ#v2-employment#n3. These links turn out to be essential for mapping between verbal and nominalized expressions (e.g. using “destroy”-”destruction”, as mentioned earlier). However, the current links do not state the semantic type of the relation (e.g., that employee#n1 is the UNDERGOER of an employ#v2 event; employment#n3 is the employ#v2 EVENT itself), which limits WordNet’s ability to help perform semantic role labeling. In addition, not being able to distinguish the semantics of the relationships can cause errors in reasoning, for example distinguishing between H1 and H2 in:

T: Detroit produces fast cars.
 H1: Detroit’s product is fast.
 H2*: Detroit’s production is fast. [NOT entailed]

T: The Zoopraxiscope was invented by Mulbridge.
 H1: Mulbridge was the inventor of the Zoopraxiscope.
 H2*: Mulbridge was the invention of the Zoopraxiscope. [NOT entailed]

To type these links, we have used a semi-automatic process: First, the computer makes a “guess” at the appropriate semantic relation based on the morphological relationship between the noun and the verb (e.g., “-er” nouns usually refer to the agent), and the location of the two synsets in WordNet’s taxonomy. Second, a human validates and corrects these, a considerably faster progress than entering them from scratch. 9 primary semantic relations (as well as 5 rarer ones) were used, namely:

agent (e.g., employ#v2-employer#n1)
 undergoer/patient (e.g., employ#v2-employee#n1)
 instrument (e.g., shred#v1-shredder#n1)
 recipient (e.g., grant#v2-grantee#n1)
 result (e.g., produce#v2-product#n2)
 body-part (e.g., adduct#v1-adductor#n1)
 vehicle (e.g., cruise#v4-cruiser#n3)
 location (e.g., bank#v3-bank#n4)
 identity/equality (eg employ#v2-employment#n1)

The resulting database of 21,000 typed links was recently completed, constituting a major new addition to WordNet in support of deep language processing. One of the surprising side results of this effort was discovering how often the normal morphological defaults (e.g., “-er” nouns refer to agents) are violated, described in more detail in Fellbaum et al. (2007). We are now in the process of incorporating the database into our software.

3.3 Core Theories

While WordNet’s glosses and links contain world knowledge about specific entities and relations, there is also more fundamental knowledge about language and the world — e.g., about space, time, and causality — which is essential for understanding many types of text, yet is unlikely to be expressed in dictionary definitions or automatically learnable. To address this need, we are also encoding by hand a number of theories to support deeper reasoning (in the style of lexical decomposition). We have axiomatized a number of abstract core theories that underlie the way we talk about events and event structure (Hobbs, 2008). Among these are theories of composite entities (things made of other things), scalar notions (of which space, time, and number are specializations), change of state, and causality. For example, in the theory of change of state, the predication $\text{change}(e1,e2)$ says there is a change of state from state $e1$ to state $e2$. The predication $\text{changeFrom}(e1)$ says there is a change out of state $e1$. The predication $\text{changeTo}(e2)$ says there is a change into state $e2$. An inference from $\text{changeFrom}(e1)$ is that $e1$ no longer holds. An inference from $\text{changeTo}(e2)$ is that $e2$ now does hold. In the theory of causality (Hobbs, 2005), the predication $\text{cause}(e1,e2)$, for $e1$ causes $e2$, is explicated. One associated inference is that if the causing happens, then the effect $e2$ happens. A defeasible inference is that not-cause-not often is the same as cause:

$$\text{not}(\text{cause}(x,\text{not}(e))) \leftrightarrow \text{cause}(x,e)$$

In the rightward direction this is of course sometimes wrong, but if we go to the trouble of saying that the negation of something was not caused, then very often it is a legitimate conclusion that the causing did happen.

We are connecting these theories with WordNet by mapping the core (5,000 most common) WordNet synsets to the theory predicates. For example, the core part of WordNet contains 450 word senses having to do with events and event structure, and we are in the process of encoding their meanings in terms of core theory predicates. For example, if x lets e happen (WordNet sense $\text{let}\#v1$), then x does not cause e not to happen:

$$\text{let}\#v1(x,e) \leftrightarrow \text{not}(\text{cause}(x,\text{not}(e)))$$

One sense of “go” is “changeTo”, as in “I go crazy”

$$\text{go}\#v4(x,e) \leftrightarrow \text{changeTo}(e)$$

(The entity x is the subject of the eventuality e .) If x frees y (the verb sense of “free”), then x causes a change to y being free (in the adjective sense of “free”):

$$\text{free}\#v1(x,y) \leftrightarrow \text{cause}(x,\text{changeTo}(\text{free}\#a1(y)))$$

Given these mappings and the core theories themselves, this is enough to answer the entailment pair:

T: The captors freed the hostage.

H: The captors let the hostage go free.

via successive application of the above axioms:

```
(part of) H interpretation → let(x,go(y,free(y)))
  ↔ not(cause(x,not(changeTo(free#a1(y))))))
  ↔ cause(x,changeTo(free#a1(y)))
  ↔ free#v1(x,y)
```

We are still in the early stages of developing this resource and have not yet evaluated it, but we have already seen a number of examples of its potential utility in the text inference problem such as above.

3.4 Scripts

Simple inference rules, such as in the above resources, provide a direct means of drawing conclusions from a few words in the input text. However, they are largely context-independent, i.e., not sensitive to the bigger picture which the surrounding text provides. Consider the following example:

```
T: A dawn bomb attack devastated a major shrine.
H: The bomb exploded.
```

In this case, it is hard to express the required knowledge (to conclude H follows from T) as simple rules (e.g., the rules “bomb → bomb explode” or “bomb attack → bomb explode” are not adequate, as we do not want H to follow from “The police destroyed the bomb.” or “The bomb attack was thwarted”). Rather, when a person reads T, he/she recognizes a complete scenario from multiple bits of evidence (possibly in multiple sentences), and integrates what is read with that scenario. This kind of top-down, expectation-driven process seems essential for creating an overall, coherent representation of text.

Although scripts are an old idea (e.g., (Schank and Abelson, 1977)) there are reasons their use may be more feasible today. First, rapid advances in paraphrasing suggests that the matching problem — deciding if some text is expressing part of a script — may be substantially eased. (Script work in the '70s required stories to be worded in exactly the right way to fire a script). Second, two new approaches for amassing knowledge are available today that were not available previously, namely automated learning from corpora, and use of Web volunteers (e.g., (Chklovski, 2005)), and may be applicable to script acquisition (Script work in the '70s typically worked with tiny databases of scripts). Finally, techniques for language processing have substantially improved, making core tasks (e.g., parsing) less problematic, and opening the possibility to easy authoring of scripts in English, followed by machine interpretation. FrameNet (Baker et al., 1998) already provides a few small scripts, but does not currently encode the complex scenarios that we would like; a vastly expanded resource would be highly useful.

We are in the early stages of exploring this avenue, encoding scripts as a list of simple English sentences, which are then automatically translated to WordNet-sense tagged logic using our software. For example, a “bombing” script looks:

```
A building is bombed by an attacker.
The attacker plants the bomb in the building.
```

The bomb explodes.
 The explosion damages or destroys the building.
 The explosion injures or kills people in the building.

In addition, some of these sentences are flagged as “salient”. If any salient sentence matches (subsumes) part of the text, then the script is triggered. When triggered, a standard graph-matching algorithm searches for the maximal overlap between the clauses in the (interpreted) script and the clauses in the text, and then the script is unified with the text according to that maximal overlap, thus asserting the additional facts contained in the script to the text under consideration. In the example earlier, the script is triggered by, and matched with, the text, thus aligning “building” with “shrine”, and asserting additional facts including (the logic representation of) “The bomb explodes” and “The bomb was planted in the building.”.

3.5 Using DIRT Paraphrases

Like others, we have also explored the use of the DIRT paraphrase database for reasoning, and we report our experiences here for comparison. The database contains 12 million rules, discovered automatically from text, of form $(X \text{ relation1 } Y) \rightarrow (X \text{ relation2 } Y)$, where relation is a path in the dependency tree/parse between constituents X and Y. Although they are noisy (informally, about 50% seem reliable), they provided some leverage for us also, for example correctly answering:

T: William Doyle works for an auction house in Manhattan.
 H*: William Doyle never goes to Manhattan. [NOT entailed]

using the DIRT rule “IF Y works in X THEN Y goes to X” combined with negation, and

T: The president visited Iraq in September.
 H: The president traveled to Iraq.

using the (slightly strange but plausible) DIRT rule “IF Y is visited by X THEN X flocks to Y” and that “flock” is a type (hyponym) of “travel”. In our experiments, DIRT rules were used 47 times (27 correctly) on our 250 example test suite. The main cause of incorrect answers was questionable/incorrect rules in the database, e.g.:

T: The US troops stayed in Iraq.
 H*: The US troops left Iraq. [NOT entailed]

was found to be entailed using the DIRT rule “IF Y stays in X THEN Y leaves X”. In addition, DIRT does not distinguish word senses (e.g., according to DIRT, shooting a person/basket implies killing the person/basket and scoring a person/basket), also contributing errors.

Despite this, the DIRT rules were useful because they go beyond just the definitional knowledge in WordNet. For example, according to DIRT “X marries Y” implies, among other things: Y marries X; X lives with Y; X kisses Y; X has a child with Y; X loves Y — all examples of plausible world knowledge. The main limitations we

found were they were noisy, did not account for word senses, and only cover one rule pattern ($X \text{ r1 } Y \rightarrow X \text{ r2 } Y$). So, for example, a rule like “X buys Y \rightarrow X pays Money” is outside the expressive scope of DIRT.

4 Preliminary Evaluation

Although this is work in progress, we have evaluated some of these augmentations using our test suite. As our ultimate goal is deeper understanding of text, we have deliberately eschewed using statistical similarity measures between T and H, and instead used abductive reasoning to create an axiom-elaborated representation of T, and then seen if it is subsumed by H. Although not using statistical similarity clearly hurts our score, in particular assuming “no entailment” when the elaborated representation of T is not subsumed by H, we believe this keeps us appropriately focused on our longer-term goal of deeper understanding of text. The results on our 250 pairs currently are:

H or \negH predicted by:	Correct	Incorrect
Simple syntax manipulation	11	3
WordNet taxonomy + morphosemantics	14	1
WordNet logicalized glosses	4	1
DIRT paraphrase rules	27	20
H or \negH not predicted:	Correct	Incorrect
(assumed not entailed)	97	72

Thus our overall score on this test suite is 61.2%. We have also run our software on the PASCAL RTE3 dataset (Giampiccolo et al., 2007), scoring 55.7% (excluding cases where no initial logical representation could be constructed due to parse/LF generation failures). In some cases, other known limitations of WordNet (eg. hypernym errors, fine-grained senses) also caused errors in our tests (outside the scope of this paper). However, the most significant problem, at least for these tests, was lack of world knowledge.

5 Conclusion

A big challenge for deep understanding of text — constructing a coherent representation of the scene it is intended to convey — is the need for large amounts of world knowledge. We have described our work-in-progress to augment WordNet in various ways so it can better provide some of this knowledge, and described some initial experiences with those augmentations, as well as with the DIRT database. Existing WordNet already provides extensive leverage for language processing, as evidenced by the large number of groups using it. The contribution of this paper is some preliminary insight into avenues and challenges for further developing this resource. Although somewhat anecdotal at this stage, our experience suggests the augmentations have promise for further improving deep language processing, and we hope will result in a significantly improved resource.

Acknowledgements This work was performed under the DTO AQUAINT program, contract N61339-06-C-0160.

References

- Adams, R., G. Nicolae, C. Nicolae, and S. Harabagiu (2007). Textual entailment through extended lexical overlap and lexico-semantic matching. In *Proc. ACL-PASCAL Workshop on Textual and Entailment and Paraphrasing*, pp. 119–124.
- Baker, C. F., C. J. Fillmore, and J. B. Lowe (1998). The Berkeley FrameNet project. In C. Boitet and P. Whitelock (Eds.), *Proc 36th ACL Conf.*, CA, pp. 86–90. Kaufmann.
- Chambers, N., D. Cer, T. Grenager, D. Hall, C. K. MacCartney, M.-C. de Marneffe, D. R. Yeh, and C. D. Manning (2007). Learning alignments and leveraging natural logic. In *Proc. ACL-PASCAL Workshop on Textual and Entailment and Paraphrasing*, pp. 165–170.
- Chklovski, T. (2005). Collecting paraphrase corpora from volunteer contributors. In *Proc 3rd Int Conf on Knowledge Capture (KCap'05)*, NY, pp. 115–120. ACM.
- Clark, P. and P. Harrison (2008, September). Boeing's NLP System and the Challenges of Semantic Representation. In J. Bos and R. Delmonte (Eds.), *Semantics in Text Processing. STEP 2008 Conference Proceedings*, Venice, Italy.
- Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press.
- Fellbaum, C., A. Osherson, and P. Clark (2007). Putting semantics into wordnet's morphosemantic links. In *Proc. 3rd Language and Technology Conference*, Poznan, Poland.
- Giampiccolo, D., B. Magnini, I. Dagan, and B. Dolan (2007). Textual entailment through extended lexical overlap and lexico-semantic matching. In *Proc. ACL-PASCAL Workshop on Textual and Entailment and Paraphrasing*, pp. 1–9.
- Graesser, A. C. (1981). *Prose Comprehension Beyond the Word*. NY: Springer.
- Gurevich, O., R. Crouch, T. King, and V. de Paiva (2006). Deverbal nouns in knowledge representation. In *Proc. FLAIRS'06*.
- Harrison, P. and M. Maxwell (1986). A new implementation of GPSG. In *Proc. 6th Canadian Conf on AI (CSCSI-86)*, pp. 78–83.
- Hobbs, J. (2005). Toward a useful notion of causality for lexical semantics. *Journal of Semantics* 22, 181–209.
- Hobbs, J. (2008). Encoding commonsense knowledge. Technical report, USC/ISI. <http://www.isi.edu/~hobbs/cksk.html>.
- Ide, N. and J. Veronis (1993). Extracting knowledge-bases from machine-readable dictionaries: Have we wasted our time? In *Proc KB&KB'93 Workshop*, pp. 257–266.

- Lenat, D. and R. Guha (1989). *Building Large Knowledge-Based Systems*. MA: Addison-Wesley.
- Miller, G. (1995). WordNet: a lexical database for english. *Comm. of the ACM* 38(11), 39–41.
- Moldovan, D. and V. Rus (2001). Explaining answers with extended wordnet. In *Proc. ACL'01*.
- Schank, R. and R. Abelson (1977). *Scripts, Plans, Goals and Understanding*. Hillsdale, NJ: Erlbaum.
- Schubert, L. and C. Hwang (1993). Episodic logic: A situational logic for NLP. In *Situation Theory and Its Applications*, pp. 303–337.