BUILDING A LARGE ONTOLOGY FOR MACHINE TRANSLATION

Kevin Knight

USC/Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292

ABSTRACT

This paper describes efforts underway to construct a large-scale ontology to support semantic processing in the PANGLOSS knowledge-base machine translation system. Because we are aiming at broad semantic coverage, we are focusing on automatic and semi-automatic methods of knowledge acquisition. Here we report on algorithms for merging complementary online resources, in particular the LDOCE and WordNet dictionaries. We discuss empirical results, and how these results have been incorporated into the PANGLOSS ontology.

1. Introduction

The PANGLOSS project is a three-site collaborative effort to build a large-scale knowledge-based machine translation system. Key components of PANGLOSS include New Mexico State University's ULTRA parser [Farwell and Wilks, 1991], Carnegie Mellon's interlingua representation format [Nirenburg and Defrise, 1991], and USC/ISI's PENMAN English generation system [Penman, 1989]. Another key component currently under construction at ISI is the PANGLOSS ontology, a large-scale conceptual network intended to support semantic processing in other PANGLOSS modules. This network will contain 50,000 nodes representing commonly encountered objects, entities, qualities, and relations.

The upper (more abstract) region of the ontology is called the Ontology Base (OB) and contains approximately 400 items that represent generalizations essential for the various PANGLOSS modules' linguistic processing during translation. The middle region of the ontology, approximately 50,000 items, provides a framework for a generic world model, containing items representing many English word senses. The lower (more specific) regions of the ontology provide anchor points for different application domains. Both the middle and domain model regions of the ontology house the open-class terms of the MT interlingua. They also contain specific information used to screen unlikely semantic and anaphoric interpretations.

The Ontology Base is a synthesis of USC/ISI's PENMAN Upper Model [Bateman, 1990] and CMU's ONTOS concept hierarchy [Carlson and Nirenburg, 1990]. Both of these high-level ontologies were built by hand, and they were merged manually. Theoretical motivations behind the OB and its current status are described in [Hovy and Knight, 1993].

The problem we focus on in this paper is the construction of the large middle region of the ontology. Because large-scale knowledge resources are difficult to build by hand, we are pursuing primarily automatic methods applied in several stages. During the first stage we created several tens of thousands of nodes, organized them into sub/superclass taxonomies, and subordinated those taxonomies to the 400-node Ontology Base. This work we describe below. Later stages will address the insertion of additional semantic information such as restrictions on actors in events, domain/range constraints on relations, and so forth.

For the major node creation and taxonomization stage, we have primarily used two on line sources of information: (1) the Longman Dictionary of Contemporary English (LDOCE)[Group, 1978], and (2) the lexical database WordNet [Miller, 1990].

2. Merging LDOCE and WordNet

LDOCE is a learner's dictionary of English with 27,758 words and 74,113 word senses. Each word sense comes with:

- A short definition. One of the unique features of LDOCE is that its definitions only use words from a "control vocabulary" list of 2000 words. This makes it attractive from the point of view of extracting semantic information by parsing dictionary entries.
- Examples of usage.
- One or more of 81 syntactic codes.
- For nouns, one of 33 semantic codes.
- For nouns, one of 124 pragmatic codes.

WordNet is a semantic word database based on psycholinguistic principles. Its size is comparable to LDOCE, but its information is organized in a completely different manner. WordNet groups synonymous word senses into single units ("synsets"). Noun senses are organized into a deep hierarchy, and the database also contains part-of links, antonym links, and others. Approximately 55% of WordNet synsets have brief informal definitions.

Each of these resources has something to offer a large-scale natural language system, but each is missing important features present in the other. What we need is a combination of the features of both.

Our most significant project to date has been to merge LDOCE and WordNet. This involves producing a list of matching pairs of word senses, e.g.:

```
   LDOCE          WORDNET

(abdomen_0_0   ABDOMEN-1)
(crane_1_2     CRANE-1)
(crane_1_1     CRANE-2)
(abbess_0_0    ABBESS-1)
(abbott_0_0    ABBOTT-1)
   ...            ...
```

Section 4 describes how we produced this list semi-automatically. Solving this problem yields several benefits:

- It allows us to taxonomize tens of thousands of LDOCE word senses and subordinate them quickly to the Ontology Base. Section 5 describes how we did this.
- It provides a syntactic and pragmatic lexicon for WordNet, as well as careful definitions.
- It groups LDOCE senses into synonyms sets and taxonomies.
- It allows us to identify and correct errors in the original resources.

3. Related Work

Our ontology is a symbolic model for fueling semantic processing in a knowledge-based MT system. We are aiming at broader coverage (dictionary-scale) than has previously been available to symbolic MT systems. Also, we are committed to automatic and semi-automatic methods of knowledge acquisition from the start. This, and the fact that we are concentrating on a particular language-processing application, distinguishes the PANGLOSS work from the CYC knowledge base [Lenat and Guha, 1990]. We also believe that dictionaries and corpora are imperfect sources of knowledge, so we still employ human effort to check the results of our semi-automatic algorithms. This is in contrast to purely statistical systems (e.g., [Brown *et al.*, 1992]), which are difficult to inspect and modify.

There has been considerable use in the NLP community of both WordNet (e.g., [Lehman *et al.*, 1992; Resnik, 1992]) and LDOCE (e.g..., [Liddy *et al.*, 1992; Wilks *et al.*, 1990]), but no one has merged the two in order to combine their strengths. The next section describes our approach in detail.

4. Algorithms and Results

We have developed two algorithms for merging LDOCE and WordNet. Both algorithms generate lists of sense pairs, where each pair consists of one sense from LDOCE and the proposed matching sense from WordNet, if any.

4.1. Definition Match

The Definition Match algorithm is based on the idea that two word senses should be matched if their two definitions share words. For example, there are two noun definitions of "batter" in LDOCE:

- (batter_2_0) "mixture of flour, eggs, and milk, beaten together and used in cooking"
- (batter_3_0) "a person who bats, esp in baseball — compare BATSMAN"

and two definitions in WordNet:

- (BATTER-1) "ballplayer who bats"
- (BATTER-2) "a flour mixture thin enough to pour or drop from a spoon"

The Definition Match Algorithm will match (batter_2_0) with (BATTER-2) because their definitions share words like "flour" and "mixture." Similarly (batter_3_0) and (BATTER-1) both contain the word "bats," so they are also matched together.

Not all senses in WordNet have definitions, but most have synonyms and superordinates. For this reason, the algorithm looks not only at WordNet definitions, but also at locally related words and senses. For example, if

synonyms of WordNet sense x appear in the definition of LDOCE sense y, then this is evidence that x and y should be matched.

Here is the algorithm:

Definition-Match

For each English word w found in both LDOCE and WordNet:

1. Let n be the number of senses of w in LDOCE.
2. Let m be the number of senses of w in WordNet.
3. Identify and stem all open-class, content words in the definitions (and example sentences) of all senses of w in both resources.
4. Let ULD be the union of all stemmed content words appearing in LDOCE definitions.
5. Let UWN be the same for WordNet, plus all synonyms of the senses, their direct superordinates, siblings, super-superordinates, as well as stemmed content words from the definitions of direct superordinates.
6. Let CW=(ULD $\cap$ UWN) - w. These are definition words common to LDOCE and WordNet.
7. Create matrix L of the n LDOCE senses and the words from CW. For all $0 \leq i < n$ and $0 \leq x <$ | CW |:

$$L[i,x] = \begin{cases} 1.00 & \text{if the definition of sense } i \\ & \text{in LDOCE contains word } x \\ 0.01 & \text{otherwise} \end{cases}$$

8. Create matrix W of the m WordNet senses and the words from CW. For all $0 \leq j < m$ and $0 \leq x <$ | CW |:

$$W[x,j] = \begin{cases} 1.00 & \text{if } x \text{ is a synonym or} \\ & \text{superordinate of sense } j \\ & \text{in WordNet} \\ 0.80 & \text{if } x \text{ is contained in the} \\ & \text{definition of sense } j \text{ or} \\ & \text{the definition of its} \\ & \text{superordinate} \\ 0.60 & \text{if } x \text{ is a sibling or} \\ & \text{super-superordinate of sense} \\ & j \text{ in WordNet} \\ 0.01 & \text{otherwise} \end{cases}$$

9. Create similarity matrix SIM of LDOCE and WordNet senses. For all $0 \leq i < n$ and $0 \leq j < m$:

$$\text{SIM}[i,j] = \left[\sum_{x=0}^{|\text{CW}|-1} (L[i,x] \cdot W[x,j]) \right] \; / \; |\,\text{CW}\,|$$

10. Repeat until SIM is a zero matrix:
    (a) Let SIM[y, z] be the largest value in the SIM matrix.
    (b) Generate matched pair of LDOCE sense y and WordNet sense z.
    (c) For all $0 \leq i < n$, set SIM[i, z] = 0.0.
    (d) For all $0 \leq j < m$, set SIM[y, j] = 0.0.

In constructing the SIM matrix the algorithm comes up with a similarity measure between each of the $n \cdot m$ possible pairs of LDOCE and WordNet senses. This measure, SIM[i, j], is a number from 0 to 1, with 1 being as good a match as possible. Thus, every matching pair proposed by the algorithm comes with a confidence factor.

Empirical results are as follows. We ran the algorithm over all nouns in both LDOCE and WordNet. We judged the correctness of its proposed matches, keeping records of the confidence levels and the degree of ambiguity present.

For low-ambiguity words (words with exactly two senses in LDOCE and two in WordNet), the results are:

confidence level	pct. correct	pct. coverage
≥ 0.0	75%	100%
≥ 0.4	85%	53%
≥ 0.8	90%	27%

At confidence levels ≥ 0.0, 75% of the proposed matches are correct. If we restrict ourselves to only matches proposed at confidence ≥ 0.8, accuracy increases to 90%, but we only get 27% of the possible matches.

For high-ambiguity words (more than five senses in LDOCE and WordNet), the results are:

confidence level	pct. correct	pct. coverage
≥ 0.0	47%	100%
≥ 0.1	76%	44%
≥ 0.2	81%	20%

Accuracy here is worse, but increases sharply when we only consider high confidence matches.

The algorithm's performance is quite reasonable, given that 45% of WordNet senses have no definitions and that many existing definitions are brief and contain misspellings. Still, there are several improvements to be made—e.g., modify the "greedy" strategy in which matches are extracted from SIM matrix, weigh rare words in definitions more highly than common ones, and/or score senses with long definitions lower than ones with short definitions. These improvements yield only slightly better results, however, because most failures are simply due to the fact that matching sense definitions have no words in common. For example, "seal" has 5 noun senses in LDOCE, one of which is:

> (seal_1_1) "any of several types of large fish-eating animals living mostly on cool seacoasts and floating ice, with broad flat limbs (FLIPPERs) suitable for swimming"

WordNet has 7 definitions of "seal," one of which is:

> (SEAL-7) "any of numerous marine mammals that come on shore to breed; chiefly of cold regions"

The Definition Match algorithm cannot see any similarity between (seal_1_1) and (SEAL-7), so it does not match them. However, we have developed another match algorithm that can handle cases like these.

4.2. Hierarchy Match

The Hierarchy Match algorithm dispenses with sense definitions altogether. Instead, it uses the various sense hierarchies inside LDOCE and WordNet.

WordNet noun senses are arranged in a deep is-a hierarchy. For example, SEAL-7 is a PINNIPED-1, which is on AQUATIC-MAMMAL-1, which is a EUTHERIAN-1, which is a MAMMAL-1, which is ultimately an ANIMAL-1, and so forth.

LDOCE has two fairly flat hierarchies. The *semantic code* hierarchy is induced by a set of 33 semantic codes drawn up by Longman lexicographers. Each sense is marked with one of these codes, e.g., "H" for human "P" for plant, "J" for movable object. The other hierarchy is the *genus sense* hierarchy. Researchers at New Mexico State University have built an automatic algorithm [Bruce and Guthrie, 1992] for locating and disambiguating genus terms (head nouns) in sense definitions. For example, (bat_1_1) is defined as "any of the several types of specially shaped wooden stick used for ..." The genus term for (bat_1_1) is (stick_1_1). As another example, the genus sense of (aisle_0_1) is (passage_0_7). The genus sense and the semantic code hierarchies were extracted automatically from LDOCE. The semantic code hierarchy is fairly robust, but since the genus sense hierarchy was generated heuristically, it is only 80% correct.

The idea of the Hierarchy Match algorithm is that once two senses are matched, it is a good idea to look at their respective ancestors and descendants for further matches. For example, once (animal_1_2) and ANIMAL-1 are matched, we can look into the respective animal-subhierarchies. We find that the word "seal" is locally unambiguous—only one sense of "seal" refers to an animal (in both LDOCE and WordNet). So we feel confident to match those seal-animal senses. As another example, suppose we know that (swan_dive_0_0) is the same concept as (SWAN-DIVE-1). We can then match their superordinates (dive_2_1) and (DIVE-3) with high confidence; we need not consider other senses of "dive."

Here is the algorithm:

Hierarchy-Match

1. Initialize the set of matches:
   (a) Retrieve all words that are unambiguous in both LDOCE and WordNet. Match their corresponding senses, and place all the matches on a list called M1.
   (b) Retrieve a prepared list of hand-crafted matches. Place these matches on a list called M2. We created 15 of these, mostly high-level matches like (person_0_1, PERSON-2) and (plant_2_1, PLANT-3). This step is not strictly necessary, but provides guidance to the algorithm.
2. Repeat until M1 and M2 are empty:
   (a) For each match on M2, look for words that are unambiguous within the hierarchies rooted at the two matched senses. Match the senses of locally unambiguous words and place the matches on M1.
   (b) Move all matches from M2 to a list called M3.
   (c) For each match on M1, look upward in the two hierarchies from the matched senses. Whenever a word appears in both hierarchies, match the corresponding senses, and place the match on M2.

(d) Move all matches from M1 to M2.

The algorithm operate in phases, shifting matches from M1 to M2 to M3, placing newly-generated matches on M1 and M2. Once M1 and M2 are exhausted, M3 contains the final list of matches proposed by the algorithm.

Again, we can measure the success of the algorithm along two dimensions, coverage and correctness:

phase	pct. correct	matches proposed
Step 1	99%	7563
Step 2(a)	94%	876
Step 2(c)	85%	530
Step 2(a)	93%	2018
Step 2(c)	83%	40
Step 2(a)	92%	99
Step 2(c)	100%	2

In the end, the algorithm produced 11,128 matches at 96% accuracy. We expected 100% accuracy, but the algorithm was foiled at several places by errors in one or another of the hierarchies. For example, (savings_bank_0_0) is mistakenly a subclass of river-bank (bank_1_1) in the LDOCE genus hierarchy, rather than (bank_1_4), the money-bank. "Savings bank" senses are matched in step 1(a), so step 2(c) erroneously goes on to match the river-bank of LDOCE with the money-bank of WordNet.

Fortunately, the Definition and Hierarchy Match algorithms complement one another, and there are several ways to combine them. Our practical experience has been to run the Hierarchy Match algorithm to completion, remove the matched senses from the databases, then run the Definition Match algorithm. The Definition Match algorithm's performance improves slightly after hierarchy matching removes some word senses. Once the high confidence definition matches have been verified, we use them as fuel for another run of the Hierarchy Match algorithm.

We have built an interface that allows a person to verify matches produced by both algorithms, and to reject or correct faulty matches. So far, we have 15,000 correct matches, with 10,000 to follow shortly. The next section describes what we do with them in our ontology.

5. The Current Ontology

The ontology currently contains 15,000 noun senses from LDOCE and 20,000 more from WordNet. Its purpose is to support semantic processing in the PANGLOSS analysis and generation modules. Because we have not yet taxonomized adjective and verb senses (see Section 6) semantic support is still very limited.

On the generation side, the PENMAN system requires that all concepts be subordinated to the PENMAN Upper Model, which is part of the Ontology Base (OB). It is difficult to subordinate tens of thousands of LDOCE word senses to the OB individually, but if we instead subordinate various WordNet hierarchies to the OB, the LDOCE senses will follow automatically via the WordNet-LDOCE merge.

Subordinating the WordNet noun hierarchy to the OB required about 100 manual operations. Each operation either merged a WordNet concept with an OB equivalent, inserted one or more WordNet concepts between two OB concepts, or attached a WordNet concept below an OB concept. The noun senses from WordNet (and their matches from LDOCE) fall under all three of the OB's primary top-level categories of **OBJECT**, **PROCESS**, and **QUALITY**. The PENMAN generator now has access to the semantic knowledge it needs to generate a broad range of English.

To support parsing, we have manually added about 20 mutual-disjoint assertions into the ontology. One of these assertions states that no individual can be both an **INANIMATE-OBJECT** and an **ANIMATE-OBJECT**, another states that **PERSON** and **NON-HUMAN-ANIMAL** are mutually disjoint, and so forth. A parser can use such information to disambiguate sentences like "this crane is my pet," where "crane" and "pet" have several senses in LDOCE (crane_1_1, a machine; crane_1_2, a bird; pet_1_1, a domestic animal; pet_1_2, a favorite person; etc.). The only pair of senses that are not mutually disjoint in our ontology is (crane_1_2)/(pet_1_1), so this is the preferred interpretation. So far, all mutual-disjoint links are between OB concepts. We plan a study of our lexicon to determine which nouns have senses that are not distinguishable on the basis of mutual-disjointness, and this will drive further knowledge acquisition of these assertions.

We are now integrating the ontology with ULTRA, the Prolog-based parsing component of the PANGLOSS translator. Although ULTRA parses Spanish input for PANGLOSS, the lexical items have already been semantically tagged with LDOCE sense keys, so no large-scale knowledge acquisition is necessary. Our first step has been to produce a Prolog version of the ontology, with inference rules for inheritance and propagation of mutual-disjoint links.

Another use of the ontology has been to help us refine LDOCE and WordNet themselves. For example, any

sample of the automatically-generated LDOCE genus-sense hierarchy has approximately 20% errors. Using our merged LDOCE-WordNet-OB ontology as a standard, we have been able to locate and fix a large number of these errors automatically.

6. Future Work

There are several items on our immediate agenda:

- Ontologize adjective, verb, and adverb senses from LDOCE. Most adjective senses either pertain to objects (e.g., atomic_1_1) or represent slot-value pairs in the ontology (e.g.,green_1_1 refers to COLOR/GREEN-COLOR as pertaining to PHYSICAL-OBJECTs). Most verb senses refer to PROCESSes, whose participants have class restrictions, and so forth. Much of this information can be mined from WordNet and LDOCE, as well as from online corpora.
- Extract a large Spanish lexicon for the ontology. We plan to use a bilingual Spanish-English dictionary (and merging techniques similar in spirit to the ones, described in this paper) in order to roughly annotate the ontology with Spanish words and phrases.
- Incrementally flesh out the ontology to improve the quality of PANGLOSS translations. We will focus on acquiring relations like SIZE, PURPOSE, PART-OF, POSTCONDITION, etc., through primarily automatic methods, including parsing of LDOCE definitions and processing corpora.

7. Acknowledgments

I would like to thank Richard Whitney for significant assistance in programming and verification. The Ontology Base was built by Eduard Hovy, Licheng Zeng, Akitoshi Okumura, Richard Whitney, and the author. I wish to express gratitude to Longman Group, Ltd., for making the machine readable version of LDOCE, 2nd edition, available to us. Louise Guthrie assisted in LDOCE extraction and kindly provided us with the LDOCE genus sense hierarchy. This work was carried out under ARPA Order No. 8073, contract MDA904-91-C-5224.

References

Bateman, J. 1990. Upper modeling: Organizing knowledge for natural language processing. In *Proc. Fifth International Workshop on Natural Language Generation, Pittsburgh, PA.*

Brown, P., V. Della Pietra, P. deSouza, J. Lai, and R. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics* 18(4).

Bruce, Rebecca and Louise Guthrie. 1992. Genus disambiguation: A study in weighted preference. In *Proceedings of the 15th International Conference on Computational Linguistics (COLING-92).*

Carlson, L. and S. Nirenburg. 1990. *World Modeling for NLP.* Tech. Rep. CMU-CMT-90-121, Center for Machine Translation, Carnegie Mellon University.

Farwell, D. and Y. Wilks. 1991. Ultra: A multilingual machine translator. In *Proceedings of the 3rd MT Summit.*

Longman Group. 1978. *Longman Dictionary of Contemporary English.* Essex, UK: Longman.

Hovy, E. and K. Knight. 1993. Motivating shared knowledge resources: An example from the pangloss collaboration. *(Submitted to: Theoretical and Methodological Issues in Machine Translation).*

Lehman, J., A. Newell, T. Polk, and R. Lewis. 1992. The rule of language in cognition. In *Conceptions of the Human Mind,* ed. G. Harman. Hillsdale, NJ: Lawrence Erlbaum. (Forthcoming).

Lenat, D. and R.V. Guha. 1990. *Building Large Knowledge-Based Systems.* Reading, MA: Addison-Wesley.

Liddy, E., W. Paik, and J. Woelfel. 1992. Use of subject field codes from a machine-readable dictionary for automatic classification of documents. In *Advances in Classification Research: Proc. 3rd ASIS SIG/CR Classification Research Workshop.*

Miller, George. 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography* 3(4). (Special Issue).

Nirenburg, S. and C. Defrise. 1991. Aspects of text meaning. In *Semantics and the Lexicon,* ed. J. Pustejovsky. Dordrecht, Holland: Kluwer.

Penman. 1989. *The Penman Documentation.* Tech. rep., USC/Information Sciences Institute.

Resnik, P. 1992. Wordnet and distributional analysis: A class-based approach to lexical discovery. In *Proc. AAAI Workshop on Statistically-Based NLP techniques.*

Wilks, Y., D. Fass, C. Guo, J. McDonald, T. Plate, and B. Slator. 1990. Providing machine tractable dictionary tools. *Machine Translation* 5.