Integration of Large-Scale Linguistic Resources in a Natural Language Understanding System

Lewis M. Norton, Deborah A. Dahl, Li Li, and Katharine P. Beals
Unisys Corporation
2476 Swedesford Road
Malvern, PA USA 19355
{norton,dahl,lli,beals}@tr.unisys.com

Abstract

Knowledge acquisition is a serious bottleneck for natural language understanding systems. For this reason, large-scale linguistic resources have been compiled and made available by organizations such as the Linguistic Data Consortium (Comlex) and Princeton University (WordNet). Systems making use of these resources can greatly accelerate the development process by avoiding the need for the developer to re-create this information.

In this paper we describe how we integrated these large scale linguistic resources into our natural language understanding system. Client-server architecture was used to make a large volume of lexical information and a large knowledge base available to the system at development and/or run time. We discuss issues of achieving compatibility between these disparate resources.

1 NL Engine

Natural language processing in the Unisys natural language understanding (NLU) system (Dahl, Norton and Scholz (1998), Dahl (1992)) is done by a natural language (NL) engine with the architecture shown in Figure 1. Processing stages include lexical lookup, syntactic parsing, semantic analysis, and pragmatic analysis. Each stage has been designed to use linguistic data such as the lexicon and grammar, which are maintained separately from the engine, and can easily be adapted to specific applications.

2 Linguistic Servers

The template NL Engine, on which all NL Engine applications are based, contains lexical information for about 3000 English words. This includes information on an exhaustive set of closed-class words -- prepositions, pronouns, conjunctions, etc. It also includes information for a few hundred of the most frequently-used words in each of the open-class word classes, the nouns, verbs, adjectives and adverbs. An NL Toolkit enables a developer to enter such information for additional words manually. Since the core vocabulary of 3000 words is insufficient for any real application, manual development could be a substantial task. Our linguistic servers are provided to greatly reduce the magnitude of that task. The servers contain the necessary information for many more words than the base system. This information can be extracted at development time, modified if appropriate (for instance, an application may not need all senses of a word), and included in the NL Engine application. The linguistic servers may or may not be present at run time of a fully-developed application (at the deployer's choice).

When information about a word is needed during processing, the available lexical resources are accessed in the following order:

1. application-specific vocabulary supplied by the developer (either manually or by extraction from the linguistic servers).
2. the core 3000-word vocabulary.
3. the linguistic servers, if present.

4. Finally, if the required information is not found in any of the linguistic resources, there are default assumptions for all linguistic information, to be described later.

There are four linguistic servers, corresponding to the four major categories of lexical information used in our system: lexicon, knowledge base, denotations, and semantics.

2.1 Lexicon Server

The lexicon server is based on Comlex, a machine-readable dictionary which was developed at New York University and distributed by the Linguistic Data Consortium (Grishman, Macleod and Wolf (1993)). Comlex contains detailed syntactic information for about 45,000 English words, including part of speech, morphological variations, lexical features, and subcategorizations.

Relatively little effort was needed to convert Comlex into a form usable by our system. A simple PERL program performed a conversion from the LISP syntax used for Comlex into Prolog, the language used for our system. Second, the features and subcategorizations represented in Comlex are encoded in terms of grammatical concepts first developed at NYU in the 1970's by Naomi Sager (Sager (1981)). The Unisys NLU system's syntactic component is based on Sager's work. As a result, little more than some name substitution was necessary to make the Comlex information usable by our system.

2.2 Knowledge Base Server

The knowledge base server is based on WordNet, a machine-readable hierarchical network of concepts which was developed and distributed by Princeton University (Miller (1990)), and on work done at the Information Sciences Institute (ISI) of the University of Southern California. Concepts in WordNet do not have names -- they are just sets of words (called *synsets*). ISI has supplied mnemonic names for the WordNet synsets and made them generally available to the WordNet community. (Examples of some of the ISI concept names can be seen in Figure 2.) The WordNet concepts correspond to real-world entities and phenomena in terms of which people understand the meanings of words. Our knowledge base server is currently concerned with only the concepts corresponding to nouns, because our system makes little use of hierarchical information about other parts of speech.[1] There are about 60,000 of these noun concepts in WordNet, including ancestor concepts which provide a taxonomy to the concept set.

Conversion of the WordNet KB was also straightforward. WordNet files in Prolog are part of the standard WordNet distribution. Therefore, the bulk of the task involved routine reformatting into the primitives of the Unisys NLU system. Our system already made use of a semantic network knowledge representation system known as M-PACK, a KL-ONE (Brachman and Schmolze (1985)) derivative which supports multiple inheritance. Our core system has a small M-PACK knowledge base, which we wanted to retain both to preserve compatibility with old applications and because it contained useful concepts which were not present in WordNet. To merge the two KBs, all we needed to do was to make each of the 11 unique beginners for WordNet noun hierarchies immediate children of appropriate concepts in our knowledge base. Making use of multiple inheritance, we also provided is-a links between selected WordNet synsets and the appropriate concepts in our small KB. For example, while our original KB contained a concept city_C, WordNet has two disjoint subtrees of cities (roughly corresponding to cities which are administrative centers such as capitals, and those which are not). By making both of these subtrees children of city_C we achieved the needed generalization, as shown in Figure 2.

2.3 Denotations Server

The denotations server, also based on WordNet and the ISI name list, provides the links between words and KB concepts, thereby integrating Comlex and WordNet. Because many nouns have multiple senses, the denotations server has over 100,000 such links for English nouns. A word is said to denote one or more concepts, according to these

[1] Our knowledge base server does contain aspect information for verb senses; this information was compiled at Unisys, not from WordNet.

links. Figure 3 illustrates this many-to-many relationship. In WordNet the senses of a word are ordered by their frequency of use in English, and our denotations server preserves this ordering. The denotations server supplies information to the NL Engine enabling it to extract from the knowledge base server the concepts denoted by the words extracted from the lexicon server. Also extracted are the ancestor concepts for the denoted concepts. Thus, for example, the NL Engine "knows" after extraction that Boston and Philadelphia are both cities.

2.4 Semantics Server

The semantics server, based on data compiled by our group at Unisys, supplies information about the semantic structure of concepts associated with English words, particularly verbs. For example, the verb *abridge* has an associated case frame consisting of an agent doing the abridging and an optional theme that is being abridged. Furthermore, in an English sentence using the verb*abridge*, the agent is typically found in the subject and the theme in the object. Words other than verbs can have similar information. The semantics server contains such information for about 4300 words, mostly verbs; the verbs account for over 60% of the verbs in Comlex.

There needs to be consistency between the information in the lexicon and semantics servers. For example, every verb which is declared to be ditransitive in Comlex should have a semantic rule mapping both the object and indirect object to distinct roles such as theme and goal. We developed a semi-automatic tool which examined every verb which had rules in the semantics server, and based on the lexical entry for that verb, added additional semantic rules to account for all of the verb's subcategorizations, or object options. These automatically fabricated rules were not always correct (the preposition*against* does not always imply an opposing force, for instance), but they were a good start. The most difficult manual task in reviewing these rules had to do with the issue of verb senses. Because verb senses are not separated in Comlex entries, the tool assumed that all the lexical subcategorizations of a verb applied to a single verb sense. When this was not the case, the semantic rules had to be divided into subsets for each individual sense, a process that we could not do automatically.

3 Default Linguistic Information

If information about a word is not found in any of our linguistic resources, the NL Engine can guess the required information. An unknown word will be assumed to be a proper noun, denoting a dynamically-created concept in the application's knowledge base, inserted as a child of our top-level concept "thing". A verb with no semantic information will be assigned roles such as agent or theme based on the syntax of the input utterance and statistical information about usage of these roles generally in other English verbs (Dahl (1993)). The default guesses are frequently sufficient for the NL Engine to make a usable interpretation of an input utterance containing an unknown word.

4 LAN Operation

Each linguistic server can be used to respond to multiple developers, or to multiple instances of a run-time NLU application. The servers can be run on separate processors (running under either Windows NT or UNIX), connected by a LAN. This minimizes the cost of utilizing the servers, which although they are relative large processes, can support many clients efficiently.

5 Evaluation

We analyzed a small corpus of 1330 sentences (on the subject of our NLU system) in order to give a quantitative description of the contribution of our lexicon and semantics servers. Our corpus contained forms of 526 distinct roots. Over 60% of these roots had definitions in our core vocabulary. Definitions for an additional 25% were extracted from the lexicon server. Analysis of the remaining 71 roots showed that a developer would have needed to enter definitions for 20 common nouns, 2 verbs, and 2 adjectives; the rest were truly proper nouns as assigned by default. The 24 roots not

covered were for the most part instances of technical jargon for our domain.[2]

For the 215 verbs in our corpus, again over 60% had semantic rules in our core NL Engine. Our semantics server contributed rules for an additional 38%, leaving our developer with the need to write rules (or rely on guessed default rules) for only 2 verbs. These results are summarized in Table 1. Thus, in this application the servers would have enabled the developer to avoid creating 132 lexical entries and 82 semantic rules. In addition, the default mechanism would have eliminated the need for manual entry of 47 more lexical entries.

	Lexicon Server	Semantics Server
in core	323 (61.5%)	131 (61%)
in server	132 (25%)	82 (38%)
not present	71 (14.5%)	2 (1%)
total	526 (100%)	215 (100%)

Table 1

Conclusion

We have successfully integrated diverse large-scale linguistic resources, both externally and internally compiled, using a client-server architecture, for use with a general-purpose natural language understanding system. The conversion of resources such as Comlex and WordNet into a format usable by our system was straightforward, and the resulting complex of resources executes without any performance problems in a multi-user environment. The task of a developer of a particular natural language application is greatly simplified by the presence of these resources.

In the future we plan to incorporate WordNet information for verbs into our KB server, and to add semantics rules for the remaining Comlex verbs into the semantics server. We also expect to augment the semantics server with semantic class constraints on the fillers of *roles* such as agent, and to create a fifth server, containing selection constraints.

[2] It is somewhat ironic that the words *database* and *parser* are not in Comlex!

References

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

Dahl D.A. (1992) *-Pundit - natural language interfaces.* In "Logic Programming in Action", G. Comyn, N.E. Fuchs, and M.J. Ratcliffe, eds., Springer-Verlag, Heidelberg, Germany, pp. 176-185.

Dahl D.A. (1993) *Hypothesizing case frame information for new verbs.* In "Principles and Prediction: The Analysis of Natural Language", M. Eid and G. Iverson, eds., John Benjamin Publishing Co., Philadelphia, Pennsylvania, pp. 175-186.

Dahl D.A., Norton L.M. and Scholz, K.W. (1998) *Commercialization of Natural Language Processing Technology.* Communications of the ACM, in press.

Grishman R., Macleod C. and Wolf S. (1993) *The Comlex syntax project.* Proceedings of the ARPA Human Language Technology Workshop, Morgan Kaufman, pp. 300-302.

Miller G. (1990) *Five Papers on WordNet.* International Journal of Lexicography.

Sager N. (1981) *Natural Language Information Processing.* Addison-Wesley, Reading, Massachusetts, 399 p.

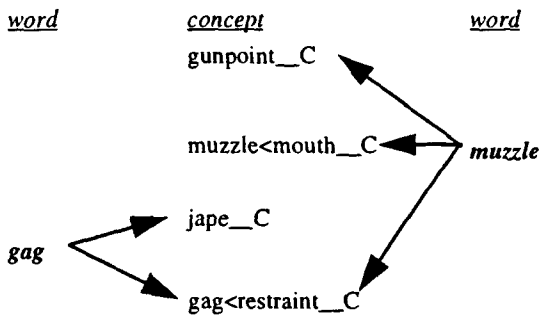

Figure 3. The *denotes* relation is many-to-many

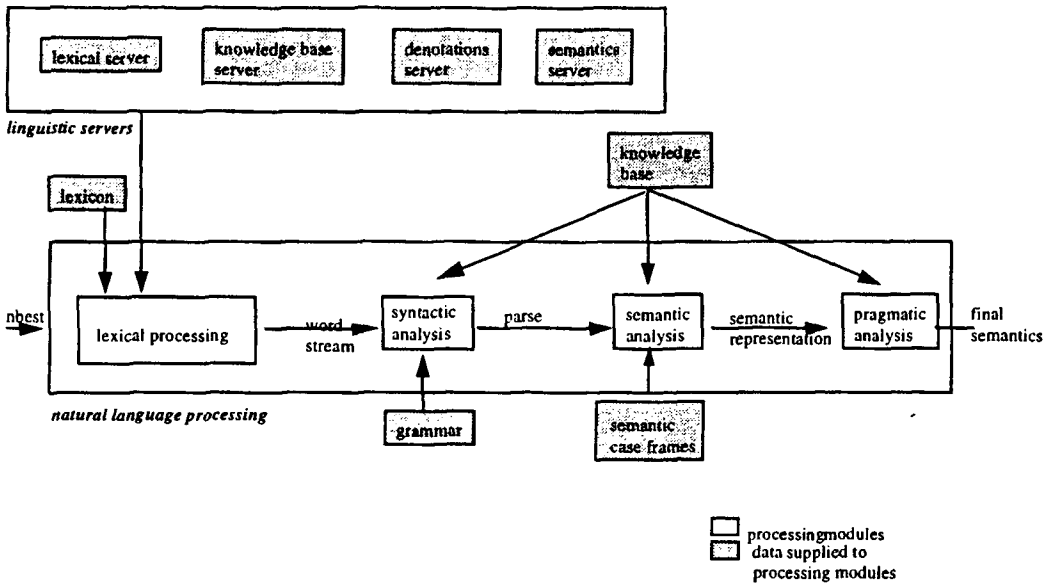

Figure 1. Overall System Architecture

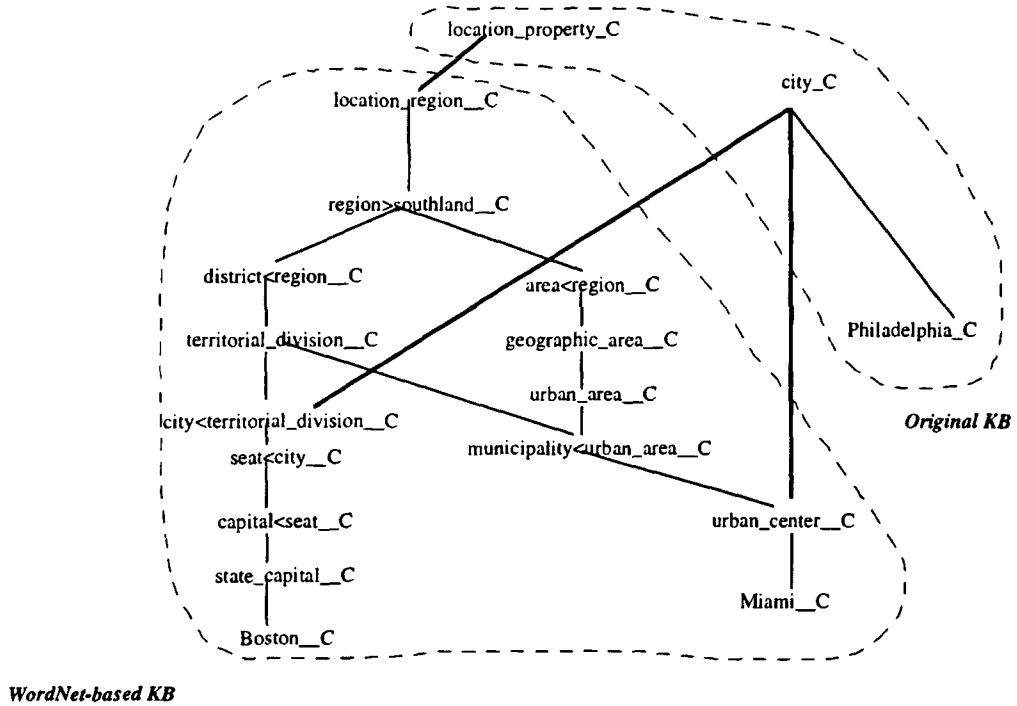

Figure 2. Integration of KB Server data with core KB

(WordNet-based KB concept names from ISI -- see text)