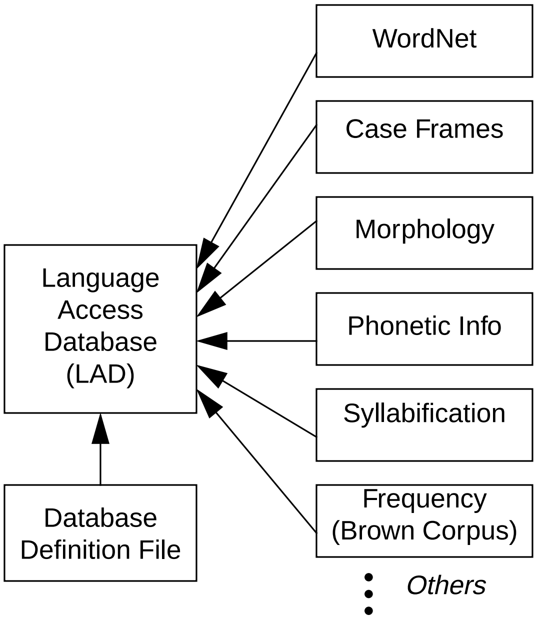

Integration of Resources

The figure above shows the overall structure of LAD. One important function of LAD is the integration of multiple lexical resources. These resources are shown on the left part of the figure. The architecture is extensible in that new lexical resources can be added without modification to the database engine. This is possible through the database definition file which defines the set of lexical resources, their location, and what attributes (e.g., frequency) they contain. Secondary lexical resources are defined as files where each record contains the word, its attribute, and an optional WordNet sense specification. The coordination of secondary databases with WordNet senses is one of the major benefits of integration. For example, the noun “bow” would be pronounced differently if it is a ornamental ribbon compared to the front of a boat.

LAD is intended to be used in several different applications. Its functionality lends itself to be a useful tool for abstracting various types of word information required by different systems. In some cases this information might not be explicitly available. For example, in systems that need verb frame information, there may be some verbs that do not have frames (e.g., pummel). By default, LAD currently retrieves a verb frame from a secondary database. In the case where the verb is not represented in the secondary database, a case frame is generated by first searching synonyms of the verb from WordNet (e.g., crush) and then checking in the secondary database for these synonyms. If that search still fails, then a case frame is generated based on the WordNet verb frame which, although it lacks detail (e.g. Somebody ---s something), would still be useful in a system that was designed to be linguistically robust.

Discussion

LAD is designed to interact with multiple lexical databases in a transparent manner. The user/system treats LAD as a single dictionary. The resulting system will be a useful tool for various AAC applications. Other applications that could benefit from LAD include a speech synthesizer needing pronunciation information and a syntactic-based word predictor using morphological information to predict correct verb forms. It is currently being tested with a semantic parser based on the reasoning principles used in Compansion. A number of enhancements are being planned that will increase the ultimate utility of the tool. This includes a compiler that will produce a more compact version of the database based on an input list of words. This will reduce the overall memory and disk space requirements when used in a practical system. In addition, while LAD is intended to be primarily used by programmers, it will also be necessary for non-technical people to enter new information into the system. For this a front-end program will be developed that will help facilitate this process.

References

[1] McHale, M. & Crowter, J. (1994) Constructing a Lexicon from a Machine Readable Dictionary. *Army Rome Laboratory Technical Report*, #RL-TR-94-178, Rome Laboratory, Griffis AFB, NY.

[2] McCoy, K., Demasco, P., Jones, M., Pennington, C., Vanderheyden, P., and Zickus, W. (1994). A communication tool for people with disabilities: Lexical semantics for filling in the pieces. In *Proceedings of ASSETS ’94.*

[3] Fillmore, C. J. (1977). The case for case reopened. In P. Cole and J. M. Sadock, editors, *Syntax and Semantics VIII: Grammatical Relations*, pp. 59-61, Academic Press, New York.

[4] Miller, Beckwith, Felbaum, Gross, Miller (1990). Introduction to WordNet: An On-line Lexical Database, *CSL Report 43*, Revised March 1993.

Acknowledgments

This work has been supported by a Rehabilitation Engineering Research Center Grant from the National Institute on Disability and Rehabilitation Research of the U.S. Department of Education (#H133E30010). Additional support has been provided by the Nemours Research Programs.

Wendy M. Zickus
Applied Science and Engineering Laboratories
1600 Rockland Road, P.O. Box 269
Wilmington, Delaware 19899 USA
Internet: zickus@asel.udel.edu

While some may contain an adequate amount of words, none of them contain the sufficient information needed to do semantic and syntactic reasoning. For instance, the word information needed for the semantic parser described above is not generally available in current systems. In addition, while there is substantial interest in the development of natural language interfaces within the general software community, there currently do not exist any lexical databases that provide both a broad coverage (in terms of numbers of words) and sufficient depth of information (e.g., case frames) for individual words (1).

Fortunately there are a variety of lexical resources available both commercially and from a variety of research laboratories. It would be advantageous to integrate these resources so that they could complement each other. This approach would allow a developer to extract desired information in a consistent, understandable and functional manner. This is the idea behind the Language Access Database (LAD).

Approach

The approach to designing LAD has been to create an implementation with C++ and Lisp[1] interfaces that allows a programmer to access several different databases (or lexical resources). The programmer is given as much or as little control as they need. For instance, they can simply query LAD about the frequency of a word and LAD will return the frequency rating found for the most generally accepted meaning of that word in some default corpus. Alternatively, if the programmer prefers, they can specify a specific "sense" of the word they are interested in and specify which corpora they would like to use.

LAD accesses several different lexical resources. The most unusual of these is an on-line dictionary/thesaurus created at Princeton University called WordNet (4). It is WordNet that contains much of the semantic information needed for intelligent AAC applications.

WordNet

At first, one might think that a computer-based lexical resource ought to be set up just like a traditional dictionary. However, this approach has some limitations. One such shortcoming is that the information stored with a word is often incomplete. When one looks up a noun, for example *platypus*, one learns that it is a semiaquatic, egg-laying mammal, but unless one is an expert on mammals, there is no way other than by looking up mammal to find out if the *platypus* has hair. Dictionaries are ordered alphabetically and not grouped semantically, therefore such searches can be cumbersome. This weakness in contemporary dictionaries demonstrates one of the major strengths of WordNet: its semantic and lexical relations. By using the WordNet on-line lexicon, it is easy to discover the attributes of a given noun by traversing the semantic relations of its superordinate term (i.e., its "parent" or category).

Another deficiency in contemporary dictionaries is the lack of information about coordinate terms (i.e., "sister" terms). Someone looking for information about other mammals would be forced to search the dictionary from beginning to end looking for terms that are classified as mammals. The prototypical lexical entry for a word points to its superordinate term, not laterally to its coordinate terms or downwards to its hyponyms (i.e., its "children" or subordinate terms). Again, these weaknesses are strengths of WordNet: its ability to reach related terms easily through its direct links to superordinate, coordinate and hyponymic terms makes searches of such information routine.

However, there are weaknesses in using only WordNet. If someone desires phonetic information, morphological forms of a word, information on non-noun/verb/adjective/adverb terms, proper nouns, or information on function words they need to go to another source. WordNet serves as a good foundation for developing a multi-purpose linguistic tool. Its breadth of coverage and sense information provides a wealth of lexical information. LAD is intended to utilize this knowledge and enhance it by using other database sources to create a centralized interface system that facilitates access to language.

Secondary Databases

Some of the other databases that LAD can access include an internally developed verb case frame database (where verb frames such as the one from our previous example are stored), a morphology database, a database containing phonetic information, a syllabification database, and statistical databases (e.g., frequency) derived from the Brown corpus and the Carterette corpus. The morphology database is important to systems like Compansion. For instance given the input *"John eat many apple"*, the system needs to be able to reason about the word *many* and change *apple* to *apples*. If the tense is present it must change *eat* to *eats* and if the tense is past, change *eat* to *ate*. Phonetic information is important for systems that need to generate speech. The statistical databases are useful for traditional AAC techniques such as word prediction.

[1] In our laboratories, we often use Lisp to develop prototypes and C++ to for commercial application development.

A LEXICAL DATABASE FOR INTELLIGENT AAC SYSTEMS

Wendy M. Zickus, Kathleen F. McCoy, Patrick W. Demasco, and Christopher A. Pennington
Applied Science and Engineering Laboratories, University of Delaware/A.I. duPont Institute
Wilmington, Delaware

Abstract

A typical non-computerized dictionary contains a wide range of information about words such as spelling, pronunciation, morphology, parts of speech, definitions, synonyms, antonyms, and other language features. The knowledge available in these dictionaries would be very useful for an intelligent AAC System; for instance, an AAC System that applies Natural Language Processing (NLP) in order to expand telegraphic messages.

This project focuses on the development of a comprehensive language database that integrates several complementary lexical resources with a single unified programming interface. This database will be used in the development of several systems that employ natural language parsing and generation techniques.

Background/Motivation

The use of Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques in the development of AAC systems and devices continues to grow both in research laboratories and, more recently, in commercial products. The use of AI/NLP methods in any application area often requires significant language knowledge such as syntax and semantics (1). Within AAC, the need to support relatively unconstrained message production (in contrast to something such as a database query) requires that this knowledge be broad as well as detailed.

One example of an intelligent AAC technique is *Compansion* (2), an approach that takes telegraphic input from a user and expands it into a syntactically and semantically well-formed sentence. The Compansion technique assumes a communication system based on words, pictures, or icons (i.e., non-spelling) and attempts to enhance the user's message production rate[1] by requiring only the selection of content words. One advantage of such a system is that it reduces the need to represent morphological information (e.g., verb inflections). This is potentially very beneficial for systems that use picture-based representations.

A major component of the Compansion system is the semantic parser which takes a set of words and attempts to fit these items into a well-formed semantic structure thus determining the intended meaning. In the current implementation, processing is non-incremental; all of the input words are taken together and a semantic representation is created which best accommodates the set of words as a whole. Generally there will be at most one word identified as the main verb in the input words; the parser must determine which semantic role is being played by the other words. Consider the processing of the input *"John break hammer"*. Once break is identified as the verb, the parser must decide which word of the input represents the agent (i.e., person or thing doing the action), which represents the theme (i.e., thing being acted upon), etc...(2). This information is represented in the semantic parser in the form of a case frame[2] for *break* (a simplified form of which is shown below):

verb:	*break*
agent:	[[human 3] [animate 2] [ergative 2]]
theme:	[[physical 3] [object 1]]
instrument:	[[tool_box 3] [tool 3] [solid 1]]
goal:	[[human 3]]
beneficiary:	[[human 3][organization 2]]
location:	[[place 4]]

The above frame indicates that the agent role is preferred to be filled by a human, but that any animate object or ergative object (e.g., a car) would also be acceptable. The theme role is preferred to be filled by a physical object, but an abstract object could also serve as a filler (although less preferred).

The basic idea of the semantic parser is to fit the non-verb words of the input into the case frame in the best way possible. In order to do this, the semantic parser must access type-information associated with each word. For instance, it must be able to tell that *John* is a human and that *hammer* is not a human but a physical object. With this information the semantic parser can reason about the words of input and generate the sentence *John breaks the hammer.*

Statement of the Problem

One of the limitations of AAC devices today is the size and information available in their dictionaries.

[1] While the Compansion techniques has been primarily described as a rate enhancement technique, it also has potential applications in helping users learn how to produce grammatical sentences

[2] A semantic representation developed by Fillmore (3).