Evaluation of Existing Reusable Resources

Costanza Navarretta

Februar 1994

ET-12 Project
Methodologies for Constructing Knowledge Bases
for Natural Language Processing Systems
Report 5

Contents

Preface

This report is part of the fourth deliverable for the ET-12 project *Methodologies for Constructing Knowledge Bases for Natural Language Processing Systems*. It is the second report of WP 3:4, 'Evaluation of Existing Reusable Resources'.

The main objectives of the project are to define a sound and general methodology for constructing knowledge bases for natural language understanding systems and to determine to which extent existing resources can be reused in this effort.

In the first phase of the project we produced a state-of-the-art survey of knowledge engineering for natural language processing systems (Report 1). In the following phase besides determining which *kinds* of knowledge should be contained in knowledge bases for natural language understanding systems and defining a working methodology for knowledge elicitation and acquisition (Report 2), we set up some criteria for possible 'support material' (Report 3).

In the present phase of the project, WP 3:4, we are refining the methodology so that the costs connected with the process of extracting knowledge from texts can be reduced and the reliability of the resulting knowledge bases can be improved. In this framework the issue of reusability of existing resources is central. This report (Report 5) contains an evaluation of existing linguistic resources as supporting material for our knowledge acquisition and elicitation methodology (described in Report 4) and an investigation of existing tools which can assist it.

The report has been written by Costanza Navarretta (CST) with the exception of the sections 2.4, 2.5, 6 and 7 which have been written by Andrei Mikheev (LTG-HCRC).

1 Introduction

Because constructing knowledge bases from scratch is an extremely resource-consuming and complex task, it is important to reduce the costs connected with it and to improve the quality of the resulting bases. The possibility of reusing existing linguistic resources is central to both issues. The aim of this report is to evaluate existing resources as material to be included in our knowledge elicitation and acquisition methodology.

In Navarretta (1993) we investigated the reusability potential of different types of linguistic resouces as sources of information to be used when constructing knowledge bases for NLU systems. Starting with a general analysis of traditional "printed" dictionaries and encyclopedias, electronic lexical resources and text corpora, we set up some criteria for possible supporting material. Of these the main criteria are the following:

1. Lexical resources must be available (by payment or by legal concession).
2. Lexical resources must be accessible by computer.
3. Lexical resources must contain both linguistic (word) and extra-linguistic (world) knowledge.
4. The information in the resource must be explicit or, if it is implicit, the nature of implicitness must be transparent for the user.
5. Resources organized for NLP systems are preferred to those composed for human use as the latter requires interpretation by a human mind.
6. Technical aspects of hardware and software, i.a. problems of storage media, the availability of access software and the way information is represented, must be considered. Resources which are portable and that contain information in a "neutral" representation form can more easily be integrated in other systems.

We found out that the linguistic resources to be further investigated because they seem to have some of the characteristics we are looking for, are machine-readable dictionaries, term banks and other electronic lexical

resources. Moreover we discovered that large general language and/or technical text corpora can be a valuable help not only when eliciting and structuring knowledge, but also when testing the coverage and the quality of the extracted information. Working with large text corpora is only viable if one has powerful tools for retrieving and manipulating the textual data, such as co-occurrence tools. We believe that techniques (mainly statistical ones) used for analysing large text corpora for NLP systems or for linguistics studies are of great relevance for our methodology (Mikheev and Navarretta 1993). Thus it is also relevant to investigate tools which implement statistical techniques.

There exist many tools for assisting the knowledge acquisition process for expert systems, but only few of them are general, i.e. have not been developed exclusively for supporting systems for solving specialized tasks. We have looked at some of these generic tools which we believe can be relevant to our work.

Another important issue in our methodology is to provide methods for commonsense knowledge conceptualization. Looking at some existing knowledge bases we have found several high-level knowledge structures that can be reimplemented for new domains. We have also looked at some existing tools which support the stage of knowledge formalization (in particular we have considered tools which support knowledge specification in CG formalism).

In the rest of this report we will first evaluate existing linguistic resources which can be used as input source of information together with the texts (machine- readable dictionaries, term banks, lexical knowledge bases, other electronic resources).

We will then evaluate tools for processing text corpora and we will describe some of the existing tools for acquiring knowledge for expert systems in general and we will evaluate tools which support knowledge formalization when using CGs.

We have made no attempt of giving an exhaustive list of all the existing resources in each of the above groups, rather we have limited ourselves to investigating those resources which are well documented in literature, which can be considered the most representative of their kind, which are readily available and/or which we found particular expedient to our work.

2 Linguistic Sources of Information

Many existing linguistic resources (e.g. machine-readable dictionaries and term banks) have been built for lexicographic use and usually presuppose a human to disambiguate them. However we believe that they can assist the knowledge engineer in the knowledge elicitation phase and in the process of characterizing words, because they have been collected and organized by people who are experts in the field of defining words and of dealing with phenomena such as homography and polysemy and, in the case of terminologists, who are also experts of the sublanguage of technical domains. Other resources such as lexical knowledge bases, specially developed for being used in NLP systems, can of course also shore up our methodology.

2.1 Machine-readable Dictionaries

Machine-readable dictionaries are useful when constructing knowledge bases for natural language systems, because they contain lots of linguistic and extra-linguistic information about words in a computerized form and because the nature of dictionary definitions is taxonomic (genus and differentia)[1]. However, the information contained in machine-readable dictionaries must be used with some care, because they (as the corresponding printed versions) have been composed after criteria that are not necessarily valid for all applications and cannot possibly contain knowledge at the granularity appropriate to all systems. Moreover they are intended to be used by people so the sense definitions they contain presuppose the existence of a human user that can interpret them, i.a. on the basis of his general background knowledge. Another problem with definitions in dictionaries is that the taxonomies derived from them can be circular.

We have chosen to evaluate the machine-readable versions of *Longman Dictionary of Contemporary English* (LDOCE) and of *Collins Cobuild, English Language Dictionary* (COBUILD), because they have many of the characteristics we are interested in[2]. First of all they are both learners' dictionaries and thus give more information about usage of words than dictionaries

[1] Special kinds of dictionaries, as thesauri, antonyms dictionaries etc. have also interest if they are available in machine readable form.

[2] In our description of the two resources we have used an unpublished paper from the Eurotra-7 project (Balkan 1990) by courtesy of Ulrich Heid.

designed for native speakers. Both dictionaries contain semantic and pragmatic information in an explicit way, or in a structured manner so that it is possible to make it explicit[3]. For the above reasons they have often been used in research projects as source for (semi)-automatically extracting semantic information about word entries.

The machine-readable version of LDOCE is available both for commercial and for research use (but with different contracts) and contains more than 41,000 entries. The electronic source of LDOCE is a typesetting tape, which is the most common medium for distributing dictionaries in machine-readable form in that tapes are simply a by-product of the process of creating the printed dictionary.

Beyond the information contained in the printed version of LDOCE (i.a. sense definitions, examples, *grammar* codes, semantic labels including geographical areas, register of discourse), the machine-readable version of LDOCE contains *subject* and *box* codes. Subject codes indicate the general context in which a word entry is most likely to appear (e.g. food, politics, language) as a help to choose the correct sense definition of a word. Box codes contain selectional restrictions on verbs, nouns and compound phrases[4] which are useful when extracting canonical structures. Another aspect which is interesting in LDOCE is that a restricted vocabulary of approx. 2000 words has been used in sense definitions, making it easier to extract taxonomic knowledge (see Boguraev and Briscoe 1989).

Some of the data on the LDOCE tape is represented in record format facilitating the conversion of the tape into a structured database. Different researchers have already implemented more strategies for converting the LDOCE tape into a database (see in Boguraev and Briscoe 1989, Briscoe 1991).

To evaluate how suitable the LDOCE ' s sense definitions are to assist the process of characterizing words in our methodology, we have used them as support source for characterizing content words from three working text corpora of different domains (cf. Bech *et al.* 1993b). We discovered that the quality of these definitions varies in relation to our particular needs. Also we believe that the "age" criterion in which the LDOCE entries are

[3]Most semantic and pragmatic information is not available in the printed version of the two dictionaries.

[4]A more complete description of these codes can be found in (Boguraev and Briscoe 1989).

ordered[5], is not the most appropriate when eliciting knowledge for NLU systems. Thus some care must be used if LDOCE sense definitions must be used as supporting material in the process of characterizing words.

COBUILD has been created from a large text corpus, the so called **Birmingham corpus**, and it is available for research and commercial users (also in this case there are two types of contract).

The machine-readable version of COBUILD is a database having word senses instead of headwords as entries (approx. 77,000). The compiled entries (records) have two forms, pink and white slips[6]. Pink slips have been recorded for each sense of the printed version's headword and contain a definition and syntactic, lexical and semantic information. White slips contain an example sentence or a citation extracted from the text corpus and detailed syntactic, lexical and semantic information related to the selected example. In the COBUILD database a pink slip is followed by one or more white slips.

COBUILD contains synonyms, antonyms and superordinates for many word entries. These types of information are very useful when collecting taxonomic knowledge and when dealing with homonymous and polynomous words. So called semantic and pragmatic fields with information not available in the printed version are also recorded in the database. In the semantic fields labels are chosen from the next level up in the hierarchical notion of lexis used for superordinates, while in the pragmatic fields explicit, concealed and implied performatives have been recorded together with the relationship of a speaker or writer to their discourse (e.g. commenting on it, structuring it etc.). The former information can help extracting taxonomic knowledge and disambiguating word senses while the latter information can help characterizing them.

Selectional preferences for verbs, adjectives and nouns are often given in a structured way by means of appropriate pronouns in the if-part of the word sense definitions, e.g.

"If you **buy** something, you obtain it by paying money for it.",

[5]The 'age' criterion is an historical criterion, i.e. the words used earliest in English are entered first.

[6]A more detailed description of the syntactic and semantic information in COBUILD can be found in (Sinclair 1987).

"If someone **buys** someone else, they get their help or services by bribing or corrupting them".

Though selectional preferences are not explicit as in LDOCE they can easily be extracted.

We have found that the COBUILD sense definitions are often very useful when characterizing word senses in our elicitation methodology. One drawback with the COBUILD definitions is that they have been selected after two different criteria ("frequency" and "age") and it is not clear to the user in which cases one criterium has been preferred.

2.2 Term banks

The definition **term bank** is used for a set of terms (belonging to one or more specialized domains) their concepts, definitions and related information stored on computerized devices. Term banks are often multilingual and constructed to support the work of domain experts and of translators. Term banks which cover the domain one has to work with can be extremely useful when constructing knowledge bases for processing texts. They can assist the knowledge engineer in the process of identifying terms and terminological multi-words (by checking which words in the actual text corpus can be found in the term bank of the appropriate domain). Term banks contain definitions and therefore they can shore up the process of characterizing terms, what is extremely important especially when the actual domain is very technical. As dictionaries they are constructed to be used by humans, but we believe that it is easier to automatically extract information from sense definitions contained in term banks because they are restricted to a specialized sublanguage so that phenomena as polysemy and homonymy are nearly absent. Term definitions can also be used for constructing the domain-specific ontology (lower levels of the general ontology).

Term banks are available in different forms (usually they are either stored on CD-ROM in which case they can be bought, or they are available on-line after subscription) and are regularly updated. Some of the most extensive term bases, available on subscription, are EURODICAUTOM (EC's on-line dictionary, multilingual) and Termium (French-English, mantained by the Terminology Directorate of the Translation bureau of the Secretary of State Derpartment of Canada).

EURODICAUTOM (Brinkhoff-Button and Folker Caroli 1990) covers all the official languages of the EU. Originally it was constructed as an online database for terminologists, translators and interpreters working at the EU, but now it is also available to users outside the EU institutions. In 1990 it contained approx. 450,000 terms for Englisk covering the following domains: agriculture, civil engineering, chemistry, communications, EC, economy, electrotechnology, health, informatics, jurisprudence, mechanical engineering, physics, transport.

In EURODICAUTOM single words, multi-words, phraseological units and abbreviations are stored. There are definitions for each entry, subject fields and synonyms, but there are no linguistic descriptions.

TERMIUM (Brinkhoff-Button and Folker Caroli 1990) is a bilingual data bank of over 500,000 terms in English and French. It is derived by a system developed by the Université de Montréal.

The entries of TERMIUM contain definitions of the terms and information about their domain, thus the terminology of a given domain can be extracted automatically. Also TERMIUM does not contain linguistic information.

2.3 Lexical knowledge bases

Electronic linguistic resources such as lexical knowledge bases[7] which are specifically constructed for different NLP systems can be very useful when eliciting and organizing knowledge from texts because a large amount of information about the meanings of the words and the way words are combined is directly available in a systematised form.

Most of the recent research on lexical knowledge bases is based on Pustejovsky's Generative Lexicon theory (Pustejovsky 1991). This theory rejects the traditional characterization of the lexicon as a static listing of word senses and proposes to consider it as a generative system in which word senses are related by logical operations defined by the well-formedness rules of the semantics. The theory proposes a formalism for describing the structures representing the semantics carried by lexical items and the rules of syntactic and semantic composition for interpreting larger expressions, including explicit methods for type coercion. The formalism includes:

[7]By lexical knowledge bases we understand lexicons in electronic form which contain an amount of world knowledge (which is usually called lexical semantic knowledge).

- The **argument structure**, i.e. the predicate argument structure for a word.
- The **event structure** which provides the identification of the particular event type for a word or a phrase.
- The **qualia structure** which partitions the aspects of a noun's meaning into formal, constitutive, agentive and telic roles.
- A **lexical inheritance network** reflecting the partition given by the qualia structure.
- **Lexical conceptual paradigms** describing sets of syntactic behaviour which correspond to lexical semantic categories. Pustejovsky distinguishes the following systems and paradigms: count/mass alternations, container/containee alternations, figure/ground reversals, product/producer diathesis, plant/fruit alternations, process/result diathesis, object/place reversals, state/thing alternations, place/people alternations.
- Generative devices for extending the logical senses of lexical items dynamically, e.g. coercion rules.

Most of the present efforts to construct lexical knowledge bases apply to some extent the Generative Lexicon theory and use machine-readable dictionaries as primary information source. Lexical knowledge bases constructed from machine-readable dictionaries must necessarily inherit some of the characteristics of the corresponding dictionaries such as their generality and the fact that the information has been collected on the basis of criteria relevant for lexicographers. However some of the problems related to machine-readable dictionaries, such as the circularity of sense definitions are not present in lexical knowledge bases in that they have been manually eliminated.

One problem related to lexical knowledge bases is that they have not been fully implemented yet (e.g. the ACQUILEX lexicon covers only a taxonomy for the food domain). We have chosen to consider them as available resources because the interest for constructing this kind of bases is very strong in the research community and we expect that they will be available in near future, providing valuable information which can easily be incorporated in NLP knowledge-based systems.

Recently Pustejovsky (1992) has proposed some techniques for acquiring the information necessary to construct sublanguage lexicons from large text corpora via syntactic and statistical corpus analysis combined with analysis strategies based on the Generative Lexicon theory. He believes that these techniques can also be used to refine the lexical structures acquired from machine-readable dictionaries and he is testing them in prototypical implementations.

The strategies/methods proposed by Pustejovsky (such as mutual information statistical techniques) can also be useful to our work.

2.4 WordNet

WordNet is a multiple-linked network of lexical items designed by Princeton University and freely available. It contains general purpose semantic information for about 53,000 words grouped in 41,500 semantic clusters.

The fundamental unit in WordNet is the synset, a grouping of words and phrases which together specify a distinguishable sense. Synsets can be linked to one another along twelve relational axes.

There are the following twelve kinds of possible relations between synsets:

- polysemy
- collocation
- verb frame
- derivation (i.e. adjectives and adverbs derived from the word)
- synonymy
- antonymy
- part-of
- substance of
- member of
- implicature

- is-a
- is sibling.

Despite of its generality, information from WordNet can be of great help in the process of sublexicon structurization.

The WordNet information can also be used for disambiguation of word senses. E.g. Feng *et al.* (1994) describe a method based on the comparison of intersections of all associated words with each sense of each word in the sentence. Senses which contribute more to the intersection are chosen as relevant. This method is also used for the reconstruction of phrasal equivalences.

2.5 Specialized Knowledge Sources

By *specialized knowledge sources* we understand electronic resources particular to a given domain, which contain more information than simple term banks. The kind of reusable information is highly dependent on the domain of interest. As a rule, these knowledge sources are well-structured, and often provide not only type-supertype dependencies but also partitive and associative links.

Although often these knowledge sources are not consistent in their choice of high-level types, their clusters of terms are usually clearly defined at middle and low levels of the hierarchy since at these levels term structurization is unambiguous and exact by nature. So even if a high-level type organization can be rejected as inconsistent with the one under construction, lower level taxons are of great importance.

In particular we have looked at the Unified Medical Language System (UMLS). UMLS is available both in paper form and as an on-line CD-ROM. It was designed to facilitate the retrieval of information from different machine readable bio-medical information sources: MEDLINE, MeSH, DXPLAIN, OMIM, PDQ, QMR, AI/RHEUM and some others. It is intended primarily for use by system developers and consists of a Metathesaurus, a Semantic Network and a Information Sources Map. All these knowledge sources are provided with a query language and search interface programs to interpret and refine user queries.

The **Metathesaurus** is a comprehensive thesaurus of the bio-medical domain. It gives the meaning of a term, its hierarchical connections (is-a relation), some other relations between terms (part-of, cause-of etc.) and some basic descriptive information. The term is a group of all strings with the same meaning and at the same time the same string (NL term) can be related to different concepts (terms).

Each concept has several attributes such as, for instance, syntactic category, semantic type, associated expression, definition, context etc.

There exist the following relations between concepts: is-a (parent, child, siblings), broader, narrower, alike, part-of, manifestation-of. Another important field represent concepts co-occurrence: qualifier, positive association, negative association etc.

The UMLS **Semantic Network** provides a consistent categorization of all the concepts represented in the Metathesaurus and specifies relations between concepts. The primary relation is the *is-a* link. The other 46 relations are grouped in four major categories: *physically related to*, *temporally related to*, *functionally related to* and *conceptually related to*.

UMLS's **Information Sources Map** provides information about electronically available sources and has a text retrieval component.

Availability of this kind of information makes KB design a lot easier since much of this expert knowledge can be reused.

2.6 Conclusion

Machine-readable dictionaries and term banks are in general available to both commercial and research use, they are stored in computerized form, they contain an amount of semantic and pragmatic information which can be extracted automatically or semi-automatically. They are not built for being used in NLP system, but presuppose the presence of a human user.

The granularity of the knowledge contained in machine-readable dictionaries cannot be adequate to all systems and applications, however we believe that machine-readable dictionaries of the type of COBUILD and LDOCE can assist the knowledge engineer in the process of extracting the higher level of general taxonomies and canonical structures. Sense definitions can also support the process of characterizing words, but some care should be taken

because dictionaries' sense definitions are not always the most adequate to this task.

Term banks covering the actual domain of discourse can support the construction of the lower levels of the ontological model and the process of characterizing terms. They can also help the knowledge engineer to acquire knowledge about specialized domains. Moreover the granularity of knowledge in term banks will usually be the right one because in most cases there is a one-to-one relation between a term and the corresponding concept. Unfortunately some of the largest available term banks (EURODICAUTOM and TERMIUM) do not contain any linguistic information which is essential to NLP systems.

Lexical knowledge bases are constructed for being used in NLP systems, they contain linguistic and extra-linguistic information in an explicit and structured way. There are research projects whos aim is building lexical knowledge bases that not only describe the language covered by general language dictionaries but also sublanguages. At present lexical knowledge bases are only available in prototypical form for research use, but we expect that they will be full-scale available in future.

Electronic resources of different nature containing some kind of semantic information about general language and/or sublanguages from specialized domains can be extremely useful when constructing knowledge bases.

3 Text Corpora and Related Tools

To work with text corpora of certain dimensions it is indispensable to have tools for automatically and quickly retrieving and manipulating the large amount of textual data they contain. Some of the tools necessary for working with text corpora are concordancing tools, lemmatisers, taggers, parsers and tools supporting statistical analysis.

3.1 Concordancing Tools

Concordancing tools are an important support to our methodology. In our work with small text corpora we have used WordCruncher[8], a tool package for preindexing DOS text files and for retrieving and manipulating the textual data they contain. It runs on PC-compatible computers under DOS. References can be looked up using a word or phrase, a list of words, two or more words in a defined context, a substring. Selected references can be displayed within windows of modifiable size. WordCruncher can generate printable KWIC-concordances, z-scores and frequency distribution reports.

The major problems with the WordCruncher package are that only documents up to 4 billion characters in size can be processed, that the documents need to be preindexed in a form which does not follow the SGML standard and that there is a limit of 100 characters for the context extracted in KWIC-concordances (which is often not enough, when dealing with referential anaphora). Concordancing tools which do not present these problems and which allow for more complex manipulation of data are available on the market.

An example of a newer and more sophisticated corpus and concordance system is CORPUS-BENCH[9] which has been developed for assisting dictionary compilation, but which can also been used in our work. CORPUS-BENCH runs on a PC under OS/2. It has many facilities for designing and setting-up large text corpora and it can quickly generate concordances, word lists and/or reports also from tagged body text. Multiple corpora can be used simultaneously. The system permits i.a. fast retrieval using filters (based on header fields and annotations and/or combination of words

[8] An *Electronic Text Corporation* product.

[9] A TEXTware product.

specified with distances) and sorting according to left/right context. It also provides some statistical reports such as mutual information to identify possible collocations, t-scores, word distribution, word frequency.

3.2 Syntactically annotated corpora

In order to apply some of the statistical strategies described in Report 4 (interim version, Mikheev and Navarretta 1993) which can be applied on large text corpora to i.a. automatically clustering words, it is necessary to have access to text corpora which are, to some extent, syntactically annotated. Lemmatisers, taggers and parsers are different types of tools, developed for annotating corpora at different complexity levels. Parsers are important when dealing with world knowledge, because they give the connections between form and meaning. However parsing is a very complex task and at present it is not possible automatically to carry out a syntactic parse of a large text corpus (There are some research groups, see the ICE initiative, that have started the enormous task of annotating existing general language corpora which then in future will be available. There already exist smaller text corpora which are tagged and/or parsed, such as the SUSANNE corpus, the BNC and the Penn Treebank). The most recent trend in the field of text corpus analysis is to build so called "bootstrapping" systems, which begin from crude processing stages (such as lemmatising) to gradually reach much greater complexity of linguistic information. A less complex task than parsing is tagging. Taggers are important for our work because many of the statistical techniques for automatically clustering words or extracting canonical structures, only require the use of tagged corpora (Mikheev and Navarretta 1993).

The general kind of specification for a tagger is that each occurrence of a word should be supplied with a word class, unambiguously. Many taggers use statistical approaches because the textual position is not always sufficient to make a clear choice among available tag alternatives. One of the first probabilistic tagging systems which has been used as example for more recent taggers, is CLAWS, Constituent-Likelihood Automatic Word-Tagging System, (Garside *et al.* 1986, McEnery 1992).

The first version of CLAWS was designed to assign a "tag" on a probabilistic basis to each of the 1,000,000 words in the Lancaster-Oslo-Bergen corpus, LOB. It achieved a 96-97 % accuracy in this task. The ultimate aim of the

developers has been to produce a tagging system which is applicable to any machine-readable corpus of modern English. CLAWS originally used a tag set of 166 tags developed from an earlier tag set used for the Brown corpus. The tags were assigned through five phases: pre-editing, tag assignment, idiom tagging, tag disambiguation and post-editing. The system consisted of three separate programs, WORDTAG, IDIOMTAG and CHAINSPROBS.

In the pre-editing phase the texts were suitably formatted (e.g. verticalization) semi-automatically. In the tag assignment phase words were tagged first by trying reference to the CLAWS knowledge base then by applying a set of rules and heuristics to the words which had not been related to a tagset. In the idiom tagging phase sequences of words which should be tagged together were sought and processed. In the tag disambiguation phase a probabilistic mechanism, built from the tagged Brown corpus and mixting 1-gram and bi-gram models, was applied to choose among candidate tags to a word. Post-editing was a manual stage for correcting wrong tags.

In the second version of CLAWS the pre-editing phase was further automatized, the tagset was expanded and the probabilistic mechanism was extended, being based on the tagged LOB corpus (by the first version of CLAWS). Another tagger VOLSUNGA, developed by DeRose (1988) has added some improvements to the CLAWS design. It has a uniform architecture, consisting of a single computer program, and contains a more efficient algorithm.

Traditional statistical approaches require large sample corpora of unambiguously pretagged texts to estimate probabilities of tag sequences and many parameters to convey these probabilities. To avoid these problems Nakamura *et al.* (1990) have proposed a neural network for word category prediction for English texts which they call NETgram. NETgram contains a core four-layer feed-forward network for bigram disambiguation. The two central layers are hidden while the remaining two are INPUT and OUTPUT layers. These layers contain 89 units which represent the complete set of possible atomic categories. Because the network is trained to guess the next word-tag as output for a given input word-tag, the two hidden layers are expected to learn some linguistic structure from the relationship between one word category and the next in the text.

The NETgram can be expanded to tackle more complex input as trigrams, 4-grams etc. E.g. a trigram model can be implemented by adding a new input layer to the original bi-gram network. The additional input layer is needed for the second word-tag pair in the trigram to be fed as input into NETgram.

A trigram network is trained with the link weight values trained by the basic bigram network as initial weights. The task of training the network is a many to many mapping problem because many alternative tags can follow each category. To reduce the time of the training process Nakamura and his colleagues use a special text corpus containing only the most probable tag sequences, calculated over a sample of 1,024 sentences.

An interesting factor is that there is a big difference in computational complexity between NETgram and traditional stochastic models[10]. Nakamura and his colleagues have evaluated the performances of the NETgram by comparing them with performances of a statistical trigram model. Testing was performed indirectly by plugging NETgram into a speech recognition system so that accuracy rates refer to the task of word recognition when tagging is used as an add-on facility. The accuracy-rates so calculated are higher than that of trigram models. Another plus with NETgram is that it tentatively provides an output also for words that do not appear as a trigram in the training corpus what is not possible with a stochastic model. Interpolating sparse trigram data using bigram training memory, NETgram obviates the inadequacy of the training data.

A tagger of the type of NETgram is interesting because it performs well also on smaller text corpora and could then be applied in our methodology on the source text corpus.

3.3 Conclusion

Many tools for i.a. generating concordances for words and phrases and for creating different statistical reports on the occurrences of words in large

[10] In a trigram model the parameters for a vocabulary C of 89 categories are $89*89*89 = 704,969$ where some of the parameters are 0 values and can be excluded with special techniques. In NETgram one can approximate $89*89*89$ transition values by using link-weight values as parameters so one gets $89*89*16$ (input-layers * lower hidden layer) $+16*16$ (the two hidden layers) $+ 16*89$ (higher hidden layer * output layer) + 121 (offset parameters) for a total of 5,193.

text corpora (of more than tens of millions of words) are already available on the market. Tools for fully automatically parsing large general language corpora do not exist yet, but systems of less complexity such as lemmatisers and taggers have been developed and can be bought. In the future it will be possible to access large parsed and tagged text corpora.

4 Hypertext Tools

Hypertext systems can support the process of organizing knowledge, because they give the possibility of dividing information into "chunks" and of linking these chunks together in any way the user of the system chooses. The user can also navigate and browse among information.

A tool developed for building term banks containing world knowledge which make use of hypertext-like facilities is CODE, Conceptually Oriented Design Environment, (Skuce and Meyer 1990, Meyer 1991). CODE is a generic knowledge engineering tool designed to assist persons to acquire, formalize, refine and access the knowledge structures of a specialized domain. It allows to construct a knowledge base which describes concepts in frame-like units called concept descriptors. Concept descriptors can both be arranged in inheritance hierarchies and in not hierarchical orderings.

The knowledge bases which CODE helps to construct are hypertext-like systems through which the user can navigate with help of a graphical interface. The graphical display provided by CODE gives the user a "picture" of the actual domain in the form of a directed graph. The arcs in the graph indicate either hierarchical or non-hierarchical (associative) relations. It is possible to focus on a part of the graph or to pan from one side to the other through very large graphs. The graphic display updates automatically when changes are made to the knowledge base and offers mechanisms for highlighting special concepts, e.g. concepts which have not been confirmed, and for comparing and contrasting knowledge substructures. The first versions of CODE have been tested in two terminological applications. The latest version of the tool is actually used to construct a bilingual prototype term bank containing world knowledge, COGNITERM.

Linster and Gaines (1990) describe an experimental environment (Hyper-KSE) that connects a knowledge acquisition tool (KSS0) with an inference tool (BABYLON) and a hypertext system (HyperCard).

KSS0 (Knowledge Support System Zero) is a tool for acquiring knowledge, using repertory-grid techniques, and for helping the user finding hidden structures (with visual feedback) and getting new insights about his own knowledge (analysis techniques). BABYLON is a hybrid knowledge representation and interpretation environment that combines rules, clauses, frames and constraints in a meta-processor so that they can interact in

the same knowledge-base. BABYLON is a generic problem-solver (i.e. it does not use one specific problem-solving method).

KSS0 is used to support the knowledge acquisition process. All constructs, values, attributes and examples can be annotated using a special HyperCard stack containing one card for each knowledge element. The card has room for text annotation and it has a link to another card that allows the input of other types of information, e.g. pictures. The KSS0 knowledge base can be then exported to a BABYLON knowledge base which can be accessed by the client through accession cards. This gives the possibility of seing what the expert meant when using an attribute or value (the cards from the KSS0 base can also be accessed in this phase). Exporting a KSS0 knowledge base into an operational environment does also allow validation and testing.

4.1 Conclusion

Hypertext and hypermedia systems can be valuable support tools to the knowledge acquisition process, because they give the possibility of declaring different paths between texts which is useful for currently testing different types of knowledge organization. CODE and environments of the type of Hyper-KSE are interesting examples of systems integrating hypertext and hypermedia facilities with knowledge acquisition tools.

5 Tools for Expert Systems

Tools developed for supporting knowledge acquisition for expert systems generally presuppose that the knowledge must be extracted from human experts. These tools can help elicit domain knowledge from experts, saving it in a form that makes it accessible for analysis, review and modification and using it to perform specific tasks. In most cases they have been built focusing on a particular problem-solving method. This restriction makes it easier to define the roles the acquired knowledge plays in finding a solution, but limits the usability potential of the tools. Examples of such tools are MOLE (Eshelman *et al.* 1987) for building heuristic problem-solving systems and KNACK (Klinker 1988) which generates expert systems for reporting tasks with the acquire-and-present problem-solving method.

In recent years a lot of progress has been made in using NL sources for building knowledge bases for expert systems. For example, the first step in the knowledge acquisition process often involves an analysis of the background literature. This allows the knowledge engineer to gain a basic understanding of the domain before embarking on interviews and other elicitation techniques. Most workbenches for the construction of knowledge based systems incorporate tools that partially automate some of the more routine parts of this knowledge acquisition process from texts. Usually these tools extract from the text corpora conceptually oriented structures that could be considered as raw material for further knowledge elicitation, organization and interpretation. The tools make use of standard information retrieval methods, combined with natural language processing techniques, text browsing facilities and hypertext methods. The aim of these tools is to extract semantically relevant data from the text on the basis of formal criteria which do not involve real text comprehension.

Obviously, none of these workbenches were designed particularly for the construction of knowledge bases for natural language processing purposes, or for the construction of linguistically anchored knowledge bases. From that point of view, their results are primitive in comparison to what we want to achieve. Nevertheless, these are also useful results which we could work on.

5.1 KRITON

Of some interest when acquiring knowledge from texts is e.g. the KRITON system (Diederich *et al.* 1987) which is a tool for automatic knowledge acquisition employing different acquisition methods to capture what the authors classify as human declarative knowledge, human procedural knowledge and static knowledge contained in natural language texts. For the elicitation of human declarative knowledge, the KRITON system contains automated interview techniques[11].The acquisition of human procedural knowledge is achieved by protocol analysis techniques. Knowledge from texts is acquired by what is called incremental text analysis. The goal structure of these different acquisition methods is an intermediate knowledge representation language.

Incremental text analysis is a tool that helps the knowledge engineer in incremental content analysis (i.e. in studying texts about the actual domain). Initially the knowledge engineer can ask for statistical information on keyword frequencies in a selected text. If a text seems useful for knowledge acquisition, the user can define the size of a text-fragment surrounding the keywords. This text fragment will then be used for the generation of basic propositions. The resulting propositional structures can be inadequate to inference processes. Therefore they are constructed in an interactive process, where possible objects and relations are shown in a menu and window system. The user picks up the correct ones with the mouse and appropriate items and knowledge structures are then set up.

The *watcher* is a component for controlling that the intermediate knowledge representation does not miss components, e.g. objects have been created, but there are not links collecting them to the domain taxonomy. At last the intermediate representation is semi-automatically (i.e. interactively with the user) translated into frames, rules and frames.

KRITON is interesting as an example of a workbench where different elicitation methods are integrated. At a more concrete level it could be possible to use in our methodology a tool similar to the incremental text analysis tool which in reality is just a tool for highlighting and retrieving text pieces using a grafic interface.

[11] A short overview of the most used interview techniques can be found in our state-of-the-art survey report (Bech *et al.* 1993a).

5.2 KADS

An expert engineering tool where different methods are used for different tasks is KADS (Breuker and Wielinga 1985). It is an interactive system which uses varying supporting functions for the knowledge engineer's activity. The functions include assistance in planning problems, data interpolation and consistency checking.

KADS is mainly based on a KL-ONE implementation in PROLOG. The rules are part of a network and the system is provided with a simple rule interpreter. KADS contains task-dependent and domain-independent information and is used for the interactive analysis of a knowledge domain. Interpretation models typical for specific problem-solving processes control the analysis.

5.3 Conclusion

We believe that it is both useful and possible to integrate our methodology in generic environments for knowledge acquisition for expert systems, and/or reuse some of the ideas behind them.

6 Commonsense Knowledge Bases

Axiomatization of commonsense knowledge is one of the main difficulties in knowledge engineering for NLU systems. Unlike domain dependent knowledge this sort of knowledge is not easily structured and formalized. Researches in this area showed that it is not possible to generate one completely reusable commonsense knowledge base (the target of the CYC project). We believe that almost any single task requires its own way of axiomatization and its own angle of view on the commonsense knowledge.

However, there exist several formal models and principles which can be reused for axiomatization of commonsense knowledge.

6.1 Built-in Types - Strings, Numbers and Sets

Two types - STRING and NUMBER - are the very foundation of each knowledge base. Although it is possible to give them a declarative representation, in many approaches they are treated as built-ins with a clearly defined procedural semantics already implemented in the computer.

It is important to mention that one knowledge base can require several different formalisms for representing different aspects of the world.

Set theory is one of the basic formal methods for conceptualization. Since we are talking about purely intensional nature of the knowledge base it is not the case that conceptual types are represented as sets of entities with particular properties as it is usual in extensional approaches. In the intensional approach sets are used to represent the plural denotation.

Of primary importance is the relation of inclusion by means of which it is possible to define such concepts as element (member) and group.

[SET]-(cardinality)→[NUMBER]
[GROUP] < [SET]
[ELEMENT]-(inclusion)→[SET]

Set theory has a well developed formal machinery and can be successfully reused .

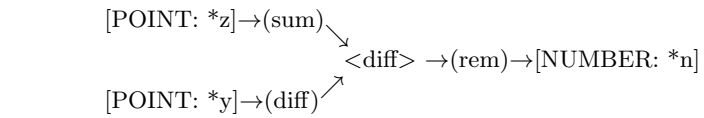
6.2 Scale Model

Gruber's (Gruber 1969) localistic theory explains conceptualization of many commonsense knowledge by using the notion of locational scale. Many commonsense fields such as, for instance, spatial, temporal, possessive etc. can be formalized using localistic theory. Later this theory was adopted by Jackendoff in his "Semantics and Cognition" (Jackendoff 1983). Conceptualization of scales plays an important role in the Hobbs' approach (Hobbs *et al.* 1988), in the CYC knowledge base (Lenat and Guha1990) and in many other projects.

Here we will present several important concepts and relations (scale functions).

The basic scale concept is the POINT. The other main concept INTERVAL can be defined as follows:

[INTERVAL: every x]-
→(low-bound)→[POINT: y]
→(up-bound)→[POINT: z]
→(meas)→[MEAS–UNIT]→(val)→[NUMBER: n]



For this definition of intervals we use two primitive types—POINT and NUMBER. Here also we use the *actor* "diff" for calculating the interval value in measure units between the points.

There are several basic relations (or scale functions) between these concepts: precedence, inclusion, overlap etc.

We can define a concept PATH as a sort of orientedinterval.

PATH < INTERVAL
[PATH]
-(from,away-from)→[POINT]
-(via)→[POINT, INTERVAL]
-(to, toward)→[POINT]

Another important concept is the VARIABLE which can move inside the scale and at any particular moment is at certain point of the scale. The following

two states BE and ORIENT denote to a location of the VARIABLE in the scale and its orientation:

[BE]
-(theme)→[VARIABLE]
-(ref)→[POINT]
-(hold)→[TIME-INTERVAL]

[ORIENT]
-(theme)→[VARIABLE]
-(along)→[PATH]
-(hold)→[TIME-INTERVAL]

We also can distinguish two kins of events in the scale:

[MOVE]
-(theme)→[VARIABLE]
-(along)→[PATH]
-(cul)→[TIME-POINT]

[STAY]
-(theme)→[VARIABLE]
-(at)→[POINT]
-(cul)→[TIME-POINT]

Many other relations and concepts can be defined using this model.

There are many different types of scales: discrete and dense, linear and clock-like, bounded and unbounded, exact and fuzzy etc. With the help of these scales it is possible to conceptualize measures, evaluations etc.

Assigning a particular meaning to a scale and a type restriction to the variable, many different fields can be conceptualized. For instance, the time field imposes a restriction on the variable to be of a temporal type and in spatial field the variable can only be of an object type. In the possession field the state POSSESS is a concretization of the state BE, etc.

7 Tools for Supporting Knowledge Formalization

Recently, the Conceptual Graph formalism gained a considerable popularity in the AI community. To support work with this formalism a special international project - the Pierce project - was launched.

The aim of this project is to fulfill the need for state-of-the-art tools in the conceptual graph community. Many people are working on particular aspects of conceptual graph theory; there are several very good implementations of subsets of the theory; but there is not as yet a robust, widely available set of tools for developing applications based on conceptual graphs.

The first public release of the Peirce Workbench demonstrated at the International Conference on Conceptual Structures in Quebec City, Canada, in August 1993. It included linear and graphical interfaces and a CG database with production rules designed to facilitate the integration of the other modules. Future versions will build on this core.

The Pierce project covers development of tools based on CG formalism in the following areas:

- *CCAT* : Conceptual Catalogs and Ontologies
- *CGC* : Programming in Conceptual Graphs With Constraints
- *DB* : Database and Retrieval
- *FACE* : Programming Interfaces for i.a. C, C++, Prolog
- *GRIP* : Graphical Interface
- *ISE* : Information Systems Engineering
- *LEARN* : Learning Mechanisms
- *LINEAR* : Linear Notation Interface
- *NLP* : Natural Language Interface: Understanding and Generation
- *PROOF* : Inference and Theorem-Proving Mechanisms
- *STDS* : Software Management and Programming Standards
- *VISION* : Vision Systems

For the purpose of knowledge engineering for NLU systems the GRIP, CGC, LINEAR and PROOF areas are of primary importance. At the same time the CCAT area can also be useful for obtaining reusable conceptualizations.

We tried to work with one of these tools – Graph Editor and Tools (GED) designed by M.Wermelinger (1991). This environment provides a nice X-Windows interface for writing statements in the CG form, it has a built-in reasoning machinery for making inferences based on facts specified in the CG form. GDE is a publicly available tool which requires Quintus X-Prolog. In spite of fairly restricted inference capabilities, we found this tool extremely useful. First the knowledge engineer can see a graphical representation of a CG statement and edit it in different ways. Also all CG statements are checked for syntactic correctness. Second, GDE supports graphs browsing with enlargement and contraction of nodes and changes of perspective of browsing. Third, it is possible to check correctness of many statements by using the inference tools.

8 Conclusion

Investigating the reusability potential of machine-readable dictionaries, term banks, lexical knowledge bases and domain specific electronic resources as supporting information source when extracting knowledge from texts, we have found that these resources can assist the knowledge engineer in the process of constructing knowledge bases for NLU systems. Machine-readable dictionaries can be useful to construct the higher levels of the ontological model, to extract canonical structures and, to some extent, to characterize words. Term banks can assist the knowledge engineer in the design of the lower levels of the ontological model and in the characterization of terms. When lexical knowledge bases will be available, it will be possible to have access to lexical semantic knowledge in a structured way. Electronic resources for specialized domains containing different kinds of semantic information can also support the process of acquiring knowledge for these domains.

Existing tools for fast retrieving and manipulating data contained in text corpora can be easily integrated in our methodology. We are also interested in tools for syntactically annotating text corpora and in large general language and/or technical corpora which are already annotated.

Hypertext and hypermedia systems, together with general systems for knowledge acquis!tion for expert systems can be used as a supporting workbench in the process of organizing the extracted knowledge and of testing and validating the contents of the resulting bases.

High level knowledge relations describing commonsense knowledge (sets, scales) can be "reused" in many domains. Tools for supporting knowledge specification in CG formalism (or in other formalisms) can be successifully used when formalizing knowledge.

In the rest of this working phase (WP 3:4) we will furthermore refine our methodology for constructing knowledge bases (Report 4, final version) also in the light of implementation tests (to be described in Report 6). In the following phase of the project we will analyse more specifically to which extent the defined methodology can be automatized, partly looking at how existing tools can be integrated in it, partly describing tools specific to this particular methodology.

References

[1] L. Balkan. Survey of Existing Reusable Lexical and Terminological Resources in the UK. Contribution for DOC-3 of the Eurotra-7 Project, October 1990.

[2] A. Bech, A. Mikheev, M. Moens, and C. Navarretta. Typology for Information Contents. ET-12 Project Report 2, EC, 1993.

[3] A. Bech, M. Moens, and C. Navarretta. Strategies in NLP knowledge engineering. ET-12 Project Report 1, EC, 1993.

[4] B. Boguraev and T. Briscoe, editors. *Computational Lexicography for Natural Language Processing*. Longman Group Limited, UK, and Wiley and Sons, Inc., NY, 1989.

[5] J. Breuker and B. Wielinga. KADS: structured knowledge acquisition for expert systems. In *Proceedings of Expert Systems and their Applications*, volume 2, pages 887–900, Avignon, France, 1985.

[6] N. Brinkhoff-Button and F. Caroli. Lexical Resources at the European Commission. EUROTRA-7: Feasibility and Project Definition Study on the Reusability of Lexical and Terminological Resources in Computerised Applications DOC-3, 1990.

[7] T. Briscoe. Lexical Issues in Natural Language Processing. ACQUILEX, ESPRIT BRA 3030 041, University of Cambridge Computer Laboratory, 1991.

[8] S. DeRose. Grammatical Category, Disambiguation by Statistical Optimization. *Computational Linguistics*, 14(1), 1988.

[9] J. Diederich, I. Ruhman, and M. May. KRITON: a knowledge- acquisition tool for expert systems. *International Journal of Man- Machine Studies*, 26(1):29–40, January 1987.

[10] L. Ehelman, D. Ehret, J. McDermott, and M. Tan. MOLE: a tenacious knowledge-acquisition tool. *International Journal of Man- Machine Studies*, 26(1):41–54, January 1987.

[11] C. Feng, T. Copeck, S. Szpacowicz, and S. Matwin. Semantic Clustering. Acquisition of Partial Ontologies from Public Domain Lexical

Sources. In *Proceedings of the 8th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, pages I, 5–17, Banff, Alberta, Canada, January – February 1994.

[12] R. Garside, R. Leech, and G. Sampson, editors. *The Computational Analysis of English.* Longman, Harlow, 1986.

[13] J.S. Gruber. *Studies in Lexical Relations.* MIT, Cambridge, 1969.

[14] J.R. Hobbs, W. Croft, T. Davies, D. Edwards, and K. Laws. The TACITUS Commonsense Knowledge Base, Draft. May 1988.

[15] R.S. Jackendoff. *Semantics and Cognition.* The MIT Press, Cambridge, Massachusetts, 1983.

[16] G. Klinker. KNACK: Sample-Driven Knowledge Acquisition Reporting Systems. In S. Marcus, editor, *Automating Knowledge Acquisition for Expert Systems*, pages 125–174. Kluwer Academic Publishers, Massachusetts, USA, 1988.

[17] D.B. Lenat and R.V. Guha. *Building Large Knowledge Bases.* Addison Wesley, 1990.

[18] M. Linster and B. Gaines. Supporting acquisition and interpretation of knowledge in a hypermedia environment. In TKE*'90 Terminology and Knowledge Engineering*, volume 1, pages 176–186, Germany, 1990. Indeks Verlag.

[19] A.M. McEnery. *Computational Linguistics - a handbook & toolbox for natural language processing.* Sigma Press, Wilmslow, United Kingdom, 1992.

[20] I. Meyer. Knowledge Management for Terminology-Intensive Applications: Needs and Tools. In J. Pustejovsky and S. Bergler, editors, *Lexical Semantics and Knowledge Representation - First* SIGLEX *Workshop*, number 627 in LNAI, pages 21–37. Springer–Verlag, Germany, 1991.

[21] A. Mikheev and C. Navarretta. Revised Methodology for Knowledge Engineering. ET-12 Project Report 4, Interim version, EC, 1993.

[22] M. Nakamura, K. Maruyama, T. Kawabata, and K. Shikano. Neural Network Approach to Word Category Prediction for English Texts. In

Proceedings of COLING, volume 3, pages 213–218, Helsinki, Finland, 1990.

[23] C. Navarretta. Criteria for 'support material'. ET-12 Project Report 3, EC, 1993.

[24] J. Pustejovsky. The Generative Lexicon. *Computational Linguistics*, 17(4):409–441, 1991.

[25] J. Pustejovsky. The Acquisition of Lexical Semantic Knowledge from Large Corpora. In *Proceedings of the Speech and Natural Language Workshop*, pages 243–248, Harriman, New York, 1992.

[26] J.M. Sinclair, editor. *Looking Up. An account of the* COBUILD *Project in Lexical computing.* Collins ELT, London, 1987.

[27] D. Skuce and I. Meyer. Computer-assisted concept analysis: an essential component of a terminologist's workstation. In TKE*'90 Terminology and Knowledge Engineering*, volume 1, pages 187–199, Germany, 1990. Indeks Verlag.

[28] M. Wermelinger. *GET - Graph Editor and Tool. Reference Manual.* CRIA/UNINOVA, 1991.