Final Report: Methodologies for Constructing Knowledge Bases for Natural Language Processing Systems

Costanza Navarretta

October 1994

ET-12 Project
Methodologies for Constructing Knowledge Bases
for Natural Language Processing Systems
Report 8

Contents

1 Introduction **2**

2 History of the Project **3**

2.1 Overview of the Work in Each Workpackage 4

3 Methodology Overview **7**

4 Perspectives for Future Research **13**

References **15**

A List of Reports and Papers of the ET-12 Project **17**

1 Introduction

This report is the eighth and final report of the ET-12 project "Methodologies for Constructing Knowledge Bases for Natural Language Processing Systems". It is the deliverable of WP 6.

In the previous phases of the project we made a state-of-the-art survey of knowledge engineering strategies for natural language processing systems and we proposed a methodology for eliciting and acquiring knowledge from natural language texts. We also described how the methodology can be shored up by existing resources and we presented a workbench for automatizing parts of the elicitation and acquisition processes. The methodology was tested on a real-life text corpus.

The aim of this report is to evaluate the project on the basis of the initial proposal, to present the results obtained and to describe the perspectives for future research. We will only present an overview of the results obtained. A more detailed description of the research done can be found in the previous reports of the project. A list of the reports and papers written in connection to the project can be found in appendix A.

This paper has been written by Costanza Navarretta (CST), with the exception of part of section 3, which is due to Andrei Mikheev (HCRC - LTG).

2 History of the Project

The primary objective of the project was to investigate systematically the task of knowledge engineering for natural language processing, focusing on the inquiry about **which** information is necessary, **how** it can be selected and **what** is the relation and interdependency between linguistic and extra-linguistic knowledge more than on the study of knowledge representation formalisms. The methodology should be generally applicable, i.e. it should not be idiosyncratic to particular systems and/or applications and should, to the extent it is possible, make (re-)use of existing resources.

At the beginning of the project, we decided to concentrate on methodologies for constructing knowledge bases for natural language understanding systems because they require the most deep and complex types of knowledge. With small modifications the proposed methodologies can be applied for constructing knowledge bases for other natural language processing systems such as information retrieval and machine translation systems.

Essentially the project has progressed along the lines given in the initial proposal. In particular the research under WP 1, WP 2, WP 5, WP 6 and WP 7 followed the original plan. Small modifications were made to WP 3 and WP 4[1] in that the tasks within the two packages were arranged in a different order and more focus was given to some aspects of the research accordingly to the results obtained in the preceding phases of the project. WP 3 and WP 4 were unified into one coherent workpackage, called WP 3:4, with the sum of the manpower allocation equal to the sum of the allocation originally foreseen for the two individual packages, i.e. 20 m/m. In WP 3:4 we gave more attention to the development of the theoretical framework and the reusability of existing linguistic tools.

[1] The revised workplan was approved by the project partners and accepted by the Commission.

The practical implementation of a knowledge base was given lower priority, mainly for practical reasons, in that it turned out that the portable version of SRI's TACITUS,[2], MINI-TACITUS, was more restricted in its capacity than foreseen and did not contain any syntactic analysis module, high-level commonsense ontology and/or primitives.

2.1 Overview of the Work in Each Workpackage

In WP 1 (October 1st 1992 – January 31st 1993) we produced a state-of-the-art survey of knowledge engineering for natural language processing systems and of methods for collecting and (re)-using existing material to support the elicitation and acquisition activities (Report 1, Bech *et al.* 1993a). Looking at the work done in the field of knowledge acquisition for NLP systems the following main tendencies were identified: bottom-up vs. top-down strategies, knowledge-based vs. lexicalist systems, corpus-supported vs. expert-based strategies. We found out that all these approaches must (and can) be combined to obtain the most promising strategy.

In WP 2 (February 1st – June 15th 1993) we emphasized the different nature between the knowledge elicitation and the knowledge acquisition processes, we determined which *kinds* of knowledge should be contained in a knowledge base for natural language understanding systems and we provided the theoretical basis for the compositional design of domain models. We distinguished between text level and word level information[3] and we found out that text corpus analysis is quite important although it has been neglected in most knowledge based systems for NLP (Report 2, Bech *et al.* 1993b). In the second report of WP 2 we determined which characteristics resources must have to support and shore up the knowledge elicitation and acquisi-

[2]The TACITUS system is a natural language understanding system developed at SRI A/S, California, in the eighties. It has been applied to several different domains, including naval equipment failure reports, naval operations reports, and terrorist reports. In the proposal it was expected that the project could adopt the original TACITUS system as its formal and implementational frame of reference, so that a knowledge base constructed applying the defined methodology could be run and tested on the system.

[3]Text level information comprises i.a. the linguistic conventions of the genre, the style, the medium, the communicative situation and competence, the discourse strategy, overt and covert connections among sentences. Word level information consists of recurrent linguistic phenomena which often need extra-linguistic knowledge to be resolved, e.g. nominal and pronominal anaphora, definite reference, compound nouns, attachment ambiguity, metonymies.

tion processes (Report 3, Navarretta 1993). To be valuable support material for knowledge acquisition strategies, linguistic resources must be available to commercial and research use and must be stored in computerized form. Moreover they must not only contain morpho-syntactic but also some semantic and (possibly) pragmatic information about words in a systematized form so that it can be extracted automatically or semi-automatically. Resources which are constructed for being used by NLP applications are of course to be preferred. General-language and technical text corpora are also of interest being first-hand material as prototypes of in-use (and/or spoken) language. Although they are good resources for shoring up the knowledge acquisition process, they must be supplied with tools for fast retrieving and manipulating the large amount of data which they contain to be useful in our framework.

The new WP 3:4 (June 15th 1993 – May 15th 1994) comprised three main tasks and included an additional deliverable (Revised Methodology for Knowledge Engineering - Interim Version). The three tasks consisted of defining the methodology for knowledge elicitation and acquisition, evaluating reusable resources and testing the methodology in practice.

First the revision of the theoretical framework outlined in Workpackage 2 was completed (Report 4, interim version, Mikheev and Navarretta 1993). The methodology for knowledge acquisition which we defined is corpus-based, application-independent and has been supplied with elicitation techniques which are specialized to the different types of knowledge to be acquired during each phase of the methodology. On the basis of this work and of the criteria set up in the previous workpackage, we evaluated the reusability of existing resources for knowledge engineering. In particular we looked at machine-readable dictionaries[4], term banks, thesauri, lexical knowledge bases, "hybrid resources"[5], text corpora and related tools for fast retrieving and manipulating the data they contain, hypermedia systems and knowledge acquisition tools for expert systems[6] (Report 5, Navarretta 1994).

[4]In particular we evaluated the machine readable versions of *Longman Dictionary of Contemporary English* and of *Collins Cobuild English Language Dictionary* because they have many of the characteristics we are interested in.

[5]By hybrid resources we mean all the electronic resources particular to a given domain which contain more information than simple term banks and/or thesauri.

[6]These tools generally presuppose that the knowledge is extracted from human experts, but many workbenches developed for supporting the construction of knowledge based systems incorporate tools that automate the extraction of conceptually oriented structures that can be considered as new material for farther knowledge elicitation, organization

The methodology was then tested on a large text corpus, the PDSs (Patient Discharge Summaries) corpus[7], belonging to the medical domain. Although the methodology was only applied on what we call "target PDS", i.e. a PDS which can be considered prototypical in relation to the kind of information which is relevant to extract from the texts, the whole corpus was used when eliciting, structuring, characterizing and refining the conceptual vocabulary of the domain (Report 6, Mikheev and Navarretta 1994a). On the basis of the experience acquired during the implementation phase we presented a new version of our methodology which is more sublanguage-oriented (Report 4, final version, Mikheev and Navarretta 1994b).

In WP 5 (May 15th – August 31st 1994) we investigated to which extent the methodology for knowledge elicitation and acquisition can be automatized and presented a toolkit for semi-automatically constructing application-independent ontologies using corpus processing methods (Report 7, Mikheev and Navarretta 1994c).

The deliverable of the last working period, WP 6 (September 1994) is the present paper, the final report.

---

and interpretation. Although their results are primitive in comparison to what we want to achieve, they present some interesting ideas where information retrieval methods are combined with NLP techniques, text browsing facilities and hypertext methods.

[7] The PDS text corpus is a large subset of a corpus used by HCRC - LTG in another EU-funded project whose aim is to extract expert information from the PDSs. The corpus we used contains 112 PDSs and consists of approx. 28,000 running words and of 2329 unique words.

3 Methodology Overview

Building on Hobbs' approach (Hobbs 1984)[8], we have developed a new methodology for acquiring and structuring background knowledge which differs in a number of aspects from the original. We have presented the methodology in two versions, the first one general (Mikheev and Navarretta 1993), the other sublanguage-oriented (Mikheev and Navarretta 1994b).

The only difference between the two versions is that in the sublanguage-oriented methodology the main concern is to arrive at a conceptualization of the domain rather than a general semantics of the words in the corpus studied. It is easier and faster to apply this version of the methodology, but the approach has the flaw of working only for one restricted sublanguage. This means that moving to a different domain requires the creation of a completely new knowledge base.

The proposed methodology is generally applicable, i.e. it is not dependent on a particular type of application and/or knowledge representation formalisms, it combines the analysis of large collections of texts with conceptual analysis (bottom-up elicitation and top-down acquisition strategies) and reuses existing resources. To which extent it is possible to reuse parts of a knowledge base obtained with the more general approach in other domains is to be proved and is still a major task in the field of knowledge engineering for natural language processing systems.

In the following we will shortly describe the proposed methodology and we will emphasize where the two versions differ.

The main principles of the methodology can be summarized as follows:

Concept-oriented approach: The lexical semantic analysis is not performed on a word by word basis but rather on the basis of abstract conceptual structures which form templates in which information from texts can be arranged. The semantics of words can then be described in terms of these conceptual structures. The knowledge engineer or computational linguist can provide a conceptualization of expressions in the sublanguage in terms of the conceptual structure of underlying

[8] Jerry Hobbs developed the so-called three-step methodology for designing application-independent knowledge bases to be run on the TACITUS system. The methodology consisted of the following steps: a) fact finding, b) fact structuring, c) formalization and adjustments.

domains. The domain terminology can then be mapped virtually unambiguously into the conceptual types. These types create the context in which other words can be defined as well.

In the sublanguage-oriented approach the primary concern is not to arrive at a conceptualization of a general semantics of the words in the studied corpus. Words are mapped into conceptual structures of the actual domain. Although the conceptualization is relative to a specific domain, in the more general approach a more wide characterization of the lexical semantics of commonsense words is to be preferred according to the strategy proposed by Hobbs (1984).

From a knowledge engineering point of view, there is a crucial difference between the two approaches: instead of determining the possible meanings for a word in the sublanguage-oriented approach we start from a structured concept-oriented representation of the domain and determine which words can express which concepts. There is no difference between the two approaches during the processing of the text. But during the knowledge engineering phase the concept- and sublanguage-oriented approach provides much more guidance for the knowledge engineer or computational linguist.

Sublanguage approach: The corpus is decomposed into a set of subdomains each of which is characterized by its own sublanguage. For each sublanguage domain specific categories are defined. Domain-specific pragmatic knowledge and specific features of a corresponding sublanguage narrow the search space during text processing.

Structured knowledge representation: An object-oriented analysis is used for a characterization of concepts. It relies on the following assumptions (more details on these assumptions can be found in Bech *et al.* 1993b):

- Use different conceptualizations for different areas of reasoning. For example, the conceptualization of scales is better done in a procedural model than in a declarative one.
- Distinguish between entities and properties, i.e. distinguish between concepts which exist by themselves, and concepts which exist only in association with other concepts. Further subcategorizations then introduce the main semantic categories.

- Distinguish extensional and intensional concepts (*e-concepts* and *i-concepts*): *e-concepts* do not have definitions and are recognized by their denoting sets; *i-concepts* have definitions and are recognized as specializations of the e-concepts.

The representation of knowledge uses both declarative and procedural means of encoding, making the representations expressive and general. This also allows for knowledge structuring at exactly the level of granularity needed to capture the semantic and pragmatic subtleties that the task requires. The investment needed to perform the acquisition and structuring of domain knowledge in this way pays off at the level of processing, allowing for more robustness, due to a reduction of ambiguities, and the possibility of stating pragmatically inspired recovery procedures to handle various types of syntactically ill-formed input.

The methodology comprises the following activities: acquiring the domain vocabulary and structuring it, characterizing conceptual schemata, characterizing domain independent words.

Vocabulary Acquisition and Compositional Structuring

The first phase of the methodology consists of acquiring the basic conceptual vocabulary for the domain by pre-processing the text corpus and roughly structuring it. Starting point for this work can be specialized dictionaries, thesauri, domain specific knowledge bases, etc.[9]

Though such knowledge repositories are very useful and important, one cannot avoid extensive lexical engineering of the underlying corpus. This phase involves the following tasks:

- An acquisition of a conceptual vocabulary by preprocessing large amounts of texts, i.e. making an extensive list of the content words in the text corpus, recognizing terms, terminological multi-words and lexical patterns[10], and collecting morphologically related words in lexical entries.

[9] For the work reported in our implementation report (Report 6), we have been using materials made available as part of the Unified Medical Language System [17] and Dorland's Medical Illustrated Dictionary [4].

[10] Lexical patterns are recurring patterns of the type {*date - day, month, year*} or {*amount - quantity, unit, measure*}.

These steps can be supported by the use of concordancing tools and linguo-statistical tools for obtaining semantic clusters of words. Then in terms of these clusters sublanguage-specific lexico-semantic patterns must be extracted.

- The vocabulary structuring which permits establishing basic domain types and nomenclature. This comprises the step of dividing the lexical entries into very general groups such as domain independent (commonsense) entries and domain specific entries (terms), and the step of splitting the so obtained groups into more specific subdomains. Word clusters are semi-automatically refined and organized into a type lattice.

- A structural compositional characterization of the basic types. Syntactic relations in lexico-semantic patterns are mapped into semantic relations, thus patterns are transformed into conceptual structures.

Text Schematization

The next phase is to arrange conceptual structures into temporal-causal schemata which support inferences and provide means for a reconstruction of implicit assumptions. This phase can be seen as a higher level conceptualization of the domain: the corpus is seen as a collection of abstract conceptual schemata each of which covers a set of fragments of the corpus. Each of these fragments are analyzed for subfragments up to the level of the domain basic types. These script-like structures encode information which is often not stated explicitly in the text. Once mentioned, it can be inherited among members of the main structure. There are certain steps that can be followed in the creation of the temporal-causal structures:

- determine general events and their subevents;

- determine temporal precedence among the subevents and assign necessity and plausability to temporally ordered pairs of subevents;

- determine causal connections between the temporally ordered pairs;

- determine thematic role fillers and their cross-correspondence and inheritance among subevents.

Characterization of Domain Independent Words

The last phase of the proposed methodology is a characterization of so-called commonsense words and phrases in the established domain framework. Commonsense words in domain oriented texts often serve as linguistic devices and indicate realization of the domain structures: they can assign modality (will, be capable of, need, seem), culmination (occur, take place, manifest, fact), resultation, etc. Generally these commonsense words are very ambiguous but in a context of domain dependent patterns they can be mapped nearly unambiguously into conceptual contents[11]. For conceptual structures of a domain we can recognize groups of local synonymy and specify their mapping onto these structures. At this stage the knowledge engineer should classify commonsense words to categories as follows:

- Commonsense words: words which correspond to such general domains as time, locations and proper names[12].
- Culminative Verbs : verbs like: "occur", "happen", "take place", etc. take their related events in the nominalized form as a subject and indicate their culmination.
- Resultators and Causators : this group of commonsense words ("result in", "cause") actualize resultation of an event and causation of the resulting state.
- Lexico-semantic Patterns: many commonsense words take their particular meaning only in a context of domain categories grouped into lexico-semantic patterns.

For each of these categories there is a predefined framework for analysis.

The methodology ends up with a characterization of not only lexical entries but also cognitive schemata of the domain (corpus).

We have supplied the methodology with a workbench for knowledge engineering which integrates computational tools and the user interface supporting the interacting processes of data extraction, data analysis and refinement

[11] In the general version of the methodology the most comprehensive characterization of a concept belonging to the commonsense domain should though be preferred.

[12] These domains are known in literature as "commonsense theories" or "clusters of commonsense knowledge" (i.a. Hayes 1979, Herzog & Rollinger eds. 1991, Hobbs & Moore eds. 1985, Lenat & Feigenbaum 1987).

(Mikheev and Navarretta 1994c). The general approach to knowledge acquisition supported by the workbench consists of a combination of methods used in knowledge engineering, information retrieval and computational linguistics. Many different tools must be incorporated in the workbench such as concordancing tools, lemmatisers, taggers, partial parsers, clustering methods, tools for accessing machine-readable dictionaries, term banks, thesauri, hypermedia, collocators, generalisers and fuzzy matchers[13]. New tools should also be added to the workbench and tools for accessing the domain specific linguistic resources must of necessity differ from domain to domain.

In order to use the information extracted/produced by the different tools is necessary to specify a common inter-module information representation format. This representation format must allow the representation of many kinds of information such as text, domain ontologies, syntactic and semantic structures.

[13]The names of the latter two tools are due to Sinclair (1994).

4 Perspectives for Future Research

The defined methodology is generally applicable and make (re)-use of existing linguistic resources and of techniques and methods developed in the fields of knowledge engineering, information retrieval and computational linguistics. To which extent the higher levels of the commonsense ontology designed with the general version of the methodology can be reused in more domains must be proved. This should be investigated by testing the methodology on more full-scale text corpora from different domains. If it is not the case that the higher levels of an ontology can be reused in processing text corpora from more different domains, the sublanguage-oriented approach should clearly be preferred.

In Mikheev and Navarretta (1994b) the hypothesis of the existence of "generic task models"[14] presented by Bylander and Chandrasekaran (1987) in connection with the KADS project was followed. To identify these generic task models and to test their generality and thus their reusability is an important issue for future research not only in the field of knowledge engineering for natural language processing but also for expert systems in general.

Because we have focused on the construction of knowledge bases for natural language understanding systems in the project, it would be useful to apply the proposed methodology for constructing knowledge bases for other natural language systems, such as machine translation systems, information retrieval systems and second-language acquisition systems.

In report 7 (Mikheev and Navarretta 1994c) we described a workbench for supporting the knowledge elicitation and acquisition process, but we did not present any technical details and suggestions about an actual implementation. This is in part because the main idea behind the workbench is that it should have an open architecture, i.e. it should be possible to incorporate in it different existing implementations and to adapt it to the different resources available for each specific domain.

[14] "Generic tasks" are basic combinations of knowledge structures and inference strategies that are powerful for solving certain kinds of problems. Thus generic tasks are models for providing a vocabulary for describing problems and for designing knowledge-based systems (or subsystems) that perform them.

Some of the tools we have described are already implemented and can be reused, other still need implementation or reimplementation in terms of the open architecture of the workbench. Implementing (and/or adapting) some of these tools is a natural sequel to the present work.

Finally we shall notice that although in the project the question of the maintenance of the knowledge base once it is constructed was not addressed, the design and implementation of a tool for knowledge base maintenance is an important issue to be considered.

References

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

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

[3] T. Bylander and B. Chandrasekaran. Generic tasks for knowledge-based reasoning: The "right" level of abstraction for knowledge acquisition. *International Journal of Man-Machine Studies*, 26:231–243, 1987.

[4] W.A.N. Dorland. *Dorland's illustrated medical dictionary, 27th edition.* Saunders, 1988.

[5] P.J. Hayes. The Naive Physics Manifesto. In D. Michie, editor, *Expert Systems in the Micro-electronic Age*, pages 242–270. Edinburgh University Press, Edinburgh, 1979.

[6] O. Herzog and C.-R. Rollinger, editors. *Text Understanding in LILOG. - Integrating Computational Linguistics and Artificial Intelligence. Final Report on the IBM Germany LILOG-Project.* Lecture Notes in Artificial Intelligence - 546. Springer–Verlag, Germany, 1991.

[7] J.R. Hobbs. Sublanguage and Knowledge. Technical Note 329, SRI, California, 1984.

[8] J.R. Hobbs and R.C. Moore, editors. *Formal Theories of the Commonsense World.* Ablex, New Jersey, 1985.

[9] D.B. Lenat and E.A. Feigenbaum. On the Thresholds of Knowledge. Technical Report ACA-AI-300-88, MCC, Austin Texas, 1988.

[10] A. Mikheev and C. Navarretta. Implementation Report. ET-12 Project Report 6, EC, 1993.

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

[12] A. Mikheev and C. Navarretta. Revised Methodology for Knowledge Engineering. ET-12 Project Report 4, Final Version, EC, 1994.

[13] A. Mikheev and C. Navarretta. Towards Automation. ET-12 Project Report 7, EC, 1994.

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

[15] C. Navarretta. Evaluation of Existing Reusable Resources. ET-12 Project Report 5, EC, 1994.

[16] J. Sinclair. Prospects for automatic lexicography. To appear in "Lexicography. Series Maior", Copenhagen, 1994.

[17] U.S. Department of Health and Human Services., National Institutes of Health. National Library of Medicine. *UMLS Knowledge Sources.*, 4th edition, 1993.

A List of Reports and Papers of the ET-12 Project

Reports:

A. Bech, M. Moens, and C. Navarretta. Strategies in NLP Knowledge Engineering. ET-12 Project Report 1, EC, 1993.

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

C. Navarretta. Criteria for 'Support Material'. ET-12 Project Report 3, EC, 1993.

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

A. Mikheev and C. Navarretta. Revised Methodology for Knowledge Engineering. ET-12 Project Report 4, Final Version, EC, 1994.

C. Navarretta. Evaluation of Existing Reusable Resources. ET-12 Project Report 5, EC, 1994.

A. Mikheev and C. Navarretta. Implementation Report. ET-12 Project Report 6, EC, 1993.

A. Mikheev and C. Navarretta. Towards Automation. ET-12 Project Report 7, EC, 1994.

C. Navarretta. Final Report: Methodologies for Constructing Knowledge Bases for NLP Systems. ET-12 Project Report 8, EC, 1994.

Papers:

A. Bech and C. Navarretta. MECKA: Methodologies for Constructing Knowledge Bases for NLP systems. In H. Albrechtsen and S. Ørnager, editors, *Knowledge Organization and Quality Management - Proceeding of the Third International ISKO Conference, Copenhagen, Denmark*, number 4 in Advances in Knowledge Organization, pages 147–153, Frankfurt/Main, Germany, June 1994. ISKO, Indeks Verlag.

A. Mikheev and M. Moens. KADS methodology for knowledge-based language processing systems. In *Proceedings of the 8th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, pages V, 1–17, Banff, Alberta, Canada, January – February 1994.

C. Navarretta. Strategier til opbygning af videnbaser. In *Bits & Bytes - Datalinguistisk foreningsårsmødet nr.3, København*, pages 7–14, Institut for Sprog og Kommunikation, Odense Universitet, Odense, 1993.

C. Navarretta. Methodologies for Knowledge Acquisition from NL Texts. In S.L. Hansen and H. Wegener, editors, *Topics in Knowledge-based NLP systems*, pages 7–17, Samfundslitteratur, Frederiksberg, 1994.