REENGINEERING LINGUISTIC RESOURCES FOR MACHINE TRANSLATION IN MEDICAL APPLICATIONS

G. Deville [a], E. Herbigniaux [a], P. Mousel [b], G. Thienpont [b]

[a] Ecole de Langues Vivantes - Facultés Universitaires de Namur
61 rue de Bruxelles - B-5000 Namur - Belgium

[b] Centre de Recherche Public - Centre Universitaire
162a avenue de la Faiencerie - L-1511 Luxembourg - Luxembourg

Abstract

We discuss some key methodological and operational aspects related to the design and development of a machine translation (MT) prototype which can be integrated in healthcare information systems. We first describe the approach adopted for collecting, formating, sampling and analyzing multilingual corpora of diagnostic expressions. The resulting generic language representation model is then presented. Finally, the prototype's architecture, its application programming interface as well as its internal structures are outlined. For economical reasons, the use of existing linguistic resources and MT technology, reengineered for an original application, was an essential requirement in this project.

1. INTRODUCTION

The present paper describes ongoing efforts in the framework of the LRE project ANTHEM[1]. The objective of ANTHEM is to develop a portable prototype of a multilingual natural language interface that allows users of healthcare information systems to enter diagnostic expressions using their own natural language and to have this input translated in whatever formal, structured or natural language. The project results from the collaboration of researchers from various disciplines: medicine, computer science and linguistics [2].

The realization of the ANTHEM prototype necessitated an extensive linguistic study of large representative corpora of medical diagnostic expressions in order to model the

---

[1] The ANTHEM project: "Advanced Natural Language Interface for Multilingual Text Generation in Healthcare" (LRE 62-007) is co-financed by the European Union within the "Linguistic Research and Engineering" program.

[2] The ANTHEM consortium is coordinated by W. Ceusters of RAMIT vzw (Ghent University Hospital) and further consists of the Institute of Modern Languages of the University of Namur (G. Deville), the IAI of the University of Saarbrücken (O. Streiter), the CRP-CU of Luxembourg (P. Mousel), the University of Liege (C. Gérardy), Datasoft Management nv — Oostende (J. Devlies) and the Military Hospital in Brussels (D. Penson).

application sublanguage and to develop the appropriate lingwares (multi-lingual lexicons and grammars). In the next section, we first discuss the medical corpus collection, formating, sampling and analysis in the prospect of the ANTHEM sublanguage modeling. Then, we briefly present the semantic representation model developed in the context of the project.

In a third section, we will focus on some operational aspects of the ANTHEM project. After a short description of the prototype's architecture, the specification of its application programming interface (API) will be presented. Finally, we will briefly sketch the prototype's internal structures.

To conclude, options for further developments will be outlined in terms of an augmentation of the lingwares and of the distribution of the prototype components.

2. A CORPUS BASED SUBLANGUAGE MODELING

In order to comply with the requirements of multidisciplinarity in the ANTHEM project, a methodology has been set up to develop and validate a medical language representation model that fulfils the following criteria:

- the model is elaborated on the basis of actual medical diagnostic expressions registred by physicians in a real-scale clinical environment (empirical validation);
- its basic concepts account for the relevant dimensions which characterize most of the sublanguage "States of Affairs" expressed in medical diagnostic expressions (descriptive adequacy);
- the model is expressed in a unification based formalism that can be directly used by the machine translation system within the ANTHEM prototype (operational validation).

These three aspects will be dealt with in this threefold section.

2.1. Corpora of Medical Diagnostic Expressions

Aiming at a representative sample of the ANTHEM application sublanguage, two corpora of medical diagnostic expressions have been collated in two different clinical environments. One source of language data in the ANTHEM project is the corpus of diagnostic expressions from the Belgian Army (referred to as the ABL corpus). The ABL corpus contains a total of 227.900 diagnostic expressions in French and Dutch, written by military doctors from 1970 to 1993, during consultations with professional servicemen and civilians in national service.

In parallel, a corpus of 12.671 Dutch and French diagnostic expressions (referred to as the MEDIDOC corpus) covering the same period as the ABL corpus, has been collected from various civilian doctors in the framework of a private consulting practice, reusing the relevant parts of the electronic medical records produced by an existing medical data managing software (MEDIDOC) that has been designed by Datasoft Management nv. As

the amount of 240.571 expressions is not manageable for linguistic analysis and modeling, a first sample of 2.343 expressions has been created on the basis of the ABL and MEDIDOC corpora, where each expression is tagged with information on its origin, language and year.

The linguistic and medical validity of these expressions was then checked respectively by linguists and medical practitioners, and a final sample of 1.362 valid expressions has been set up (i) for the elaboration of the lingwares as well as (ii) for the testing of the ANTHEM prototype. A detailed account of the methodological principles for the elaboration of the corpora of diagnostic expressions in the ANTHEM project is given in [Deville & Herbigniaux 94].

2.2. A Case Grammar Oriented Model of the Medical Diagnostic Sublanguage

Following [Dik 89] we refer to the underlying semantic representation of a diagnostic expression as a predication. A predication is a structure that consists of a predicate with an adequate number of terms functioning as arguments of that predicate. Terms are phrases (i.e. noun phrases or prepositional phrases) used to refer to entities or a set of entities in the conceptual world of a given sublanguage (a sublanguage being defined here as a set of expressions referring to a limited and well-defined application domain and used for a specific function). A predicate (or head) is a noun phrase (i.e. noun or adjective) capturing semantic properties of or semantic relations between its arguments. A predication refers to a Sublanguage world Configuration. A Sublanguage world Configuration (hence SwC) is a cluster of sublanguage conceptual entities that are expressed in terms of their mutual relationships [Deville 89].

In the case of ANTHEM, the sublanguage conceptual entities are defined as minimal units that not only refer to atomic (e.g. arm) or complex objects (e.g. left hand palm), but also mainly to states (e.g. inflammation), relations (e.g. part of) or to a lesser extent actions (e.g. ingestion) and processes (e.g. to fall). More precisely, the terms of the ANTHEM application sublanguage refer to objects, and predicates to states, relations and actions/processes.

Predicates (or heads) are selected from a finite set of semantic types. A semantic type captures the prototypical semantic and combinatorial properties shared by a set of predicates. Most of the semantic types used in the ANTHEM representation model are inherited from the Systematized Nomenclature of Human and Veterinary Medicine (SNOMED), that is a widely used standard in the field of medical terminology [CAP 93].

In a predication, the relation between the predicate and its arguments is specified by means of a case. A case is the expression of a prototypical semantic function or role fulfilled by a predicate's term with regard to the semantic class from which that predicate derives. The case frame of a semantic type is the sequence of required cases for the definition of the set of SwCs represented by that semantic type. As a predicate and its arguments refer to a particular SwC, a semantic type with its associated case frame refers, on a more prototypical conceptual level, to a class of SwCs. Such a higher level

structure specifies the semantic roles of the arguments of the derived predicate, in relation to its corresponding semantic type.

A predication can be extended by means of one or more peripheral arguments. Peripheral arguments are not constitutive of the definition of a SwC but express the spatio-temporal setting of that SwC, the secondary entities participating in the SwC or gives information on the manner or conditions in which the SwC takes place. As opposed to central arguments, the semantic functions of peripheral arguments are not necessary to define a set of SwCs in terms of a semantic type with its associated case frame.

2.3. Implementation of the Linguistic Model

The above described model is being implemented in the form of Prolog rules. These rules form one part of the lingwares, referred to as the ANTHEM grammars (French, Dutch and German). The ANTHEM prototype contains one grammar module for each language.

The grammar rules are applied to objects which are the lexical entries corresponding to the terms used in the ANTHEM application domain. These lexical entries form the second part of the lingwares, referred to as the ANTHEM lexicons. The ANTHEM prototype includes one lexicon for each language, plus a common lexicon which enables the translation of terms from one language into another.

3. A PROTOTYPE BASED ON EXISTING MACHINE TRANSLATION TECHNOLOGY

3.1. Architecture of the ANTHEM Prototype

The aim of the ANTHEM project is to develop a software module which can easily be integrated in host applications in the domain of medical data registration. The function of the ANTHEM prototype is to translate medical diagnostic statements expressed in a natural source language either into a natural target language, a medical coding system or a semantic representation [Ceusters et al. 94].

To achieve its task, the module relies on existing machine translation technology. It presents a well defined, simple Application Programming Interface (API) to any host application written in C and uses translation and encoding services offered by specialized programs. The architecture of the overall system is shown on figure 1.

The following components of the prototype can be identified:

- The medical host application can be any application in a healthcare environment which requires automatic translation of medical diagnostic expressions. During the evaluation phase of the ANTHEM project, we will integrate the system in two different medical applications: (i) one developed by Datasoft Management nv and devoted to the management of a medical practitioner's patient files, (ii) the other developed at the Military Hospital in Brussels and used for statistical purposes.

- The ANTHEM API consists of a set of C functions which provide well defined services to the medical host application and hide implementation details of the underlying software layers. The interface and the internal structure of this component will be further detailed in the following sections.
- The CAT2 system is a unification-based general purpose machine translation system developed at the Institute of Applied Informatics in Saarbrücken [Streiter et al. 94]. In the ANTHEM project, CAT2 is being adapted for translating natural language diagnostic expressions or for transforming these expressions into a language-independent semantic representation. The adopted methodology for "right-sizing" CAT2 has been discussed in section 2.
- An expert system (ES), currently under development at the laboratory for research in advanced medical informatics and telematics of the Ghent University Hospital, transforms a language independent semantic representation of a diagnostic expression into standardized medical codes (ICD-10). We will not further describe this module, given its embryonic state at this stage of the project.

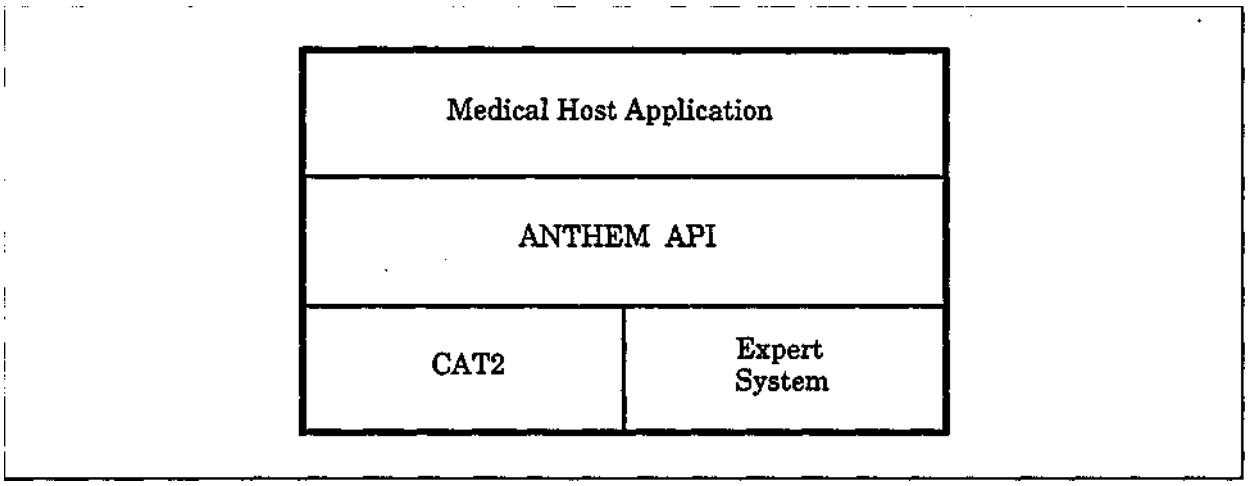

figure 1 —Architecture of the ANTHEM prototype

We have currently developed a UNIX version of the prototype which runs on Sun workstations. It is integrated in a dummy application (fig. 2), based on the OSF/Motif toolkit, which allows easy and efficient testing of the API and of the underlying modules. In its present state, the prototype includes the Dutch and French lingwares developed on the basis of a first test sample of 200 diagnostic expressions.

In the next two sections some insights into the ANTHEM API will be given.

3.2. The ANTHEM API

We wanted to keep the API as simple as possible to facilitate the task of the developers who envisage to integrate the ANTHEM prototype into either existing or new medical host

applications. As the API has been implemented in C, it essentially consists of an header file and a function library [Mousel & Thienpont 94].

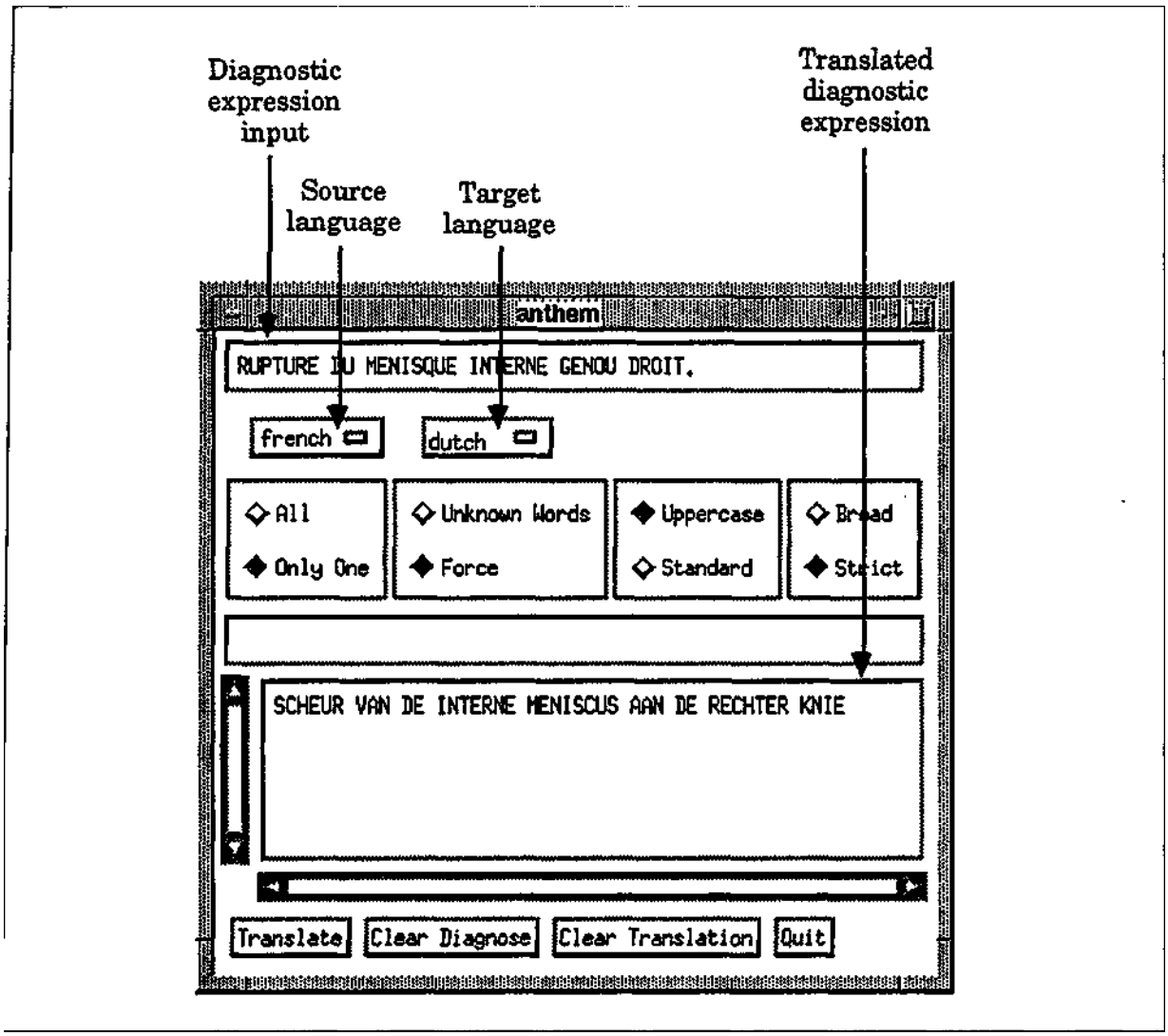

figure 2 — Dummy application user interface

The header file contains several type and macro definitions that are of little interest here.

The library contains only five public C functions which offer all the necessary services to the medical host application:

- The first call to the ANTHEM API has to be a call to a function whose purpose is to initialize the ANTHEM module. This function may only be called once.
- The last call to ANTHEM must be a call to a function whose task is to clean up the ANTHEM module. Similarly, this function may only be called once.

- The three remaining functions may be called at any time and in any order between the calls to the two previous ones. One function provides the list of possible source languages, another gives the list of possible target languages whereas the third is called by the host application to translate a diagnostic statement expressed in a source language into a target language.

In order to use ANTHEM's translation and encoding services, the medical host application must include the header file and must be linked with the library.

3.3. Implementation of the ANTHEM API

The API hides all the details of the underlying layers to the medical host application. As the objective of the ANTHEM project is to capitalize on existing technology, we reengineered proven MT software to provide the actual translation and encoding services:

- For the translation of diagnostic expressions, ANTHEM makes use of the CAT2 translation system, which is written in Prolog and has been designed for interactive use.
- For the ICD-10 coding of diagnostic expressions, ANTHEM uses an ES also written in Prolog and designed for interactive use.

Thus, the API has to behave like a human user with respect to the translation and the ICD-10 coding systems, which means that the interface sends commands to CAT2 and the ES and waits for their results. Communication between the API and both components is based on the pipe mechanism. For the sake of portability, the API exclusively uses standard POSIX primitives to manipulate the pipes. Figure 3 shows the internals of the ANTHEM prototype.

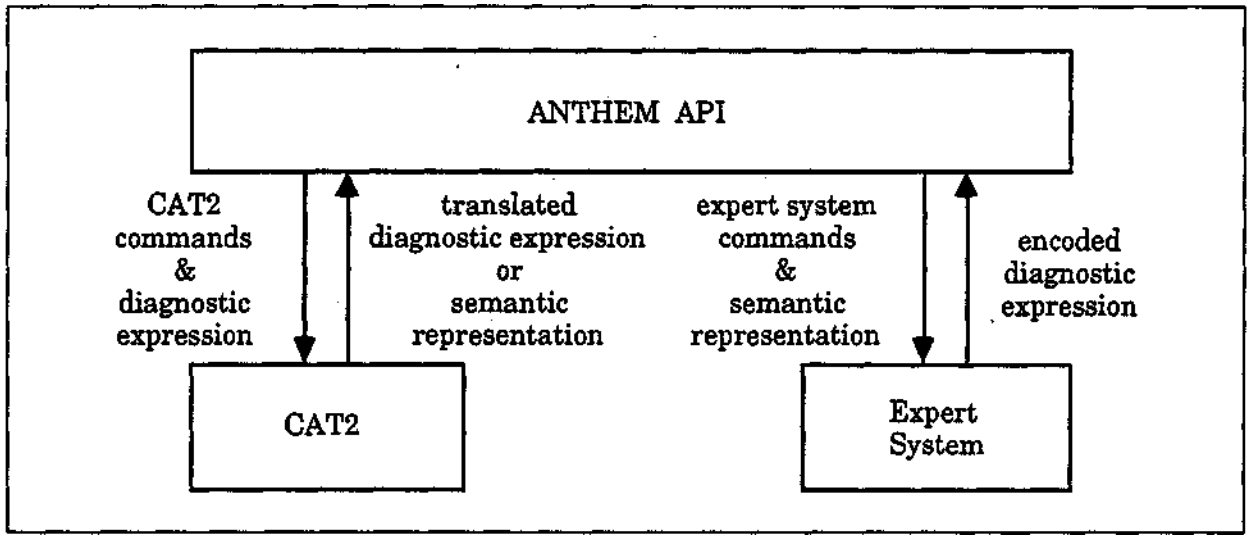

figure 3 — Communication between ANTHEM components

4. CONCLUSION

The paper discussed some key methodological and operational aspects related to the design of a medical MT system.

We have presented how, starting from the study of corpora of diagnostic expressions, a language model has been elaborated and described in a unification based formalism that is suitable for the development of the ANTHEM lingwares.

Then we have shown how the ANTHEM prototype uses existing MT technology to provide translation and encoding services to medical host applications. A stand alone version of the prototype has been implemented on Sun workstations and integrated in a dummy application.

The following issues will be dealt with in future work:

- The prototype will be tested against large volumes of additional linguistic material and the lingwares will be augmented appropriately.
- A client/server version based on the DCE (Distributed Computing Environment) framework in a mixed PC/UNIX environment is under development and a stand alone version on a PC platform is envisaged.
- Finally, both client/server and stand alone versions will be integrated in real scale medical applications for on site evaluation purposes.

5. REFERENCES

[CAP 93] College of American Pathologists, SNOMED International, Introduction, 1993

[Ceusters et al.] Ceusters W., Deville G., Mousel P., Streiter O. and Thienpont G., Functional Specification of the ANTHEM Prototype, ANTHEM Deliverable n. Dl-1, 1994

[Dik 89] Dik S., A Theory of Functional Grammar, Foris, Dordrecht, 1989

[Deville 89] Deville G., Modelization of Task-Oriented Utterances in a Man-Machine Dialogue System, Ph.D. Thesis, Universitaire Instelling Antwerpen, 1989

[Deville & Herbigniaux 94] Deville G. and Herbigniaux E., Methodological Principles for the Elaboration of Multilingual Corpora of Medical Diagnostic Expressions, ANTHEM Deliverable n. D2-2 - Part I, 1994

[Mousel & Thienpont 94] Mousel P. and Thienpont G., Technical Specification of the ANTHEM Prototype, ANTHEM Deliverable n. Dl-2, 1994

[Streiter et al. 94] Streiter O., Haller J., Sharp R., Schmitt-Wigger A. & Pease C., Aspects of a Unification Based Multilingual System for Computer Aided Translation, Proceedings of the 14th international conference "Avignon '94", May 30th - June 3rd 1994