

**Towards an Automatic Translation of Medical Terminology and Texts
into Arabic**

Catherine Pease
IAI
Martin-Luther-Straße 14
D-66111 Saarbrücken
Germany
Tel. +49 681 389510
Fax +49 681 3895140
email : cath@iai.uni-sb.de

Abdelaziz Boushaba
UFRL Case 7003
Universite Paris 7
TALANA
2, Place Jussieu
F-75251 Paris cedex 05
France
Tel. +33 1 44 275696
Fax +33 1 44 277919
email : zohir@linguist.jussieu.fr

**Translation in the Arab World
King Fahd Advanced School of Translation
Tangier, November 27-30, 1996
Proceedings**

Towards an Automatic Translation of Medical Terminology and Texts into Arabic

Catherine Pease
IAI
Martin–Luther–Straße 14
D–66111 Saarbrücken
Germany
email : cath@iai.uni-sb.de

Abdelaziz Boushaba
UFRL Case 7003
Universite Paris 7, TALANA
2, Place Jussieu
F–75251 Paris cedex 05
France
email : zohir@linguist.jussieu.fr

Abstract

The Aramed project, financed by the INCO programme of the European Commission, is developing a system which translates medical classifications (based on the SNOMED medical codes) from English to Arabic, and German and English medical texts into Arabic. Its aim is to provide a usable tool for automatic translation in the area of medicine in the Arab world. It also provides the basis of a German–Arabic MT system. The system consists of two main components: an Arabic morphological generator (NALG) and a transfer, constraint and unification–based MT system (CAT2), developed as a sideline to the Eurotra MT project. In this paper we shall explain in detail how a language like Arabic can be relatively easily included in a MT system which was originally developed for European languages, and how general linguistic and terminological knowledge is integrated into one framework of analysis which, using a few principles of syntactic and semantic composition, can translate both general language and medical texts.

1 INTRODUCTION TO ARAMED

The ARAMED project is financed by the CEC (Commission of the European Communities) as part of the INCO programme – (a programme for INternational COoperation with Third Countries and International Organisations). It involves three partners: The Institute for Applied Information Science (IAI), Saarbrücken as main contractor, and The Electronics Research Institute (ERI), Cairo and TALANA (of the Department of Linguistics in the University of Paris 7), Paris as partners. The aim is to translate medical terminology (coded according to the internationally recognised SNOMED (Systemised Nomenclature of Medicine) codes written by the College of American Pathologists), and medical texts from English and German into Arabic. This aim addresses a very real need in the Arab world, where the area of medicine is dominated by the French and English languages. It is hoped that by translating part of the SNOMED codes into Arabic, the writing of an official version of these codes in Arabic will be initiated, which will bring the Arab world further into the international world of medicine. It will also be a step forward for the many scientists (in this case physicians)

who are trying to encourage the use of Arabic in science and technology. The translation is performed by the CAT2 MT system, and the syntactic representation in Arabic is then passed to the NALG morphological generator for generation of the Arabic target text with its full derivational and inflectional information.

2 THE CAT2 MT SYSTEM

The CAT2 Machine Translation System was first developed in 1987 as a sideline to Eurotra. From 1992 to 1995 the CAT2 system underwent a pilot study, with support from the CEC, of industrial validation in cooperation with a major software company to test its possible application for commercial purposes. At present the system is being developed in various projects: the most important is a project called Multilint, which was started in July 1995 by the German Federal Ministry for Economics and a major German automobile company and aims to translate car repair messages from German into English. Other projects include ANTHEM – which is developing a system to analyse medical diagnostic expressions in Dutch and French and translate them into German, French and Dutch and into a semantic representation used for automatic encoding. The Aramed project started in January 1996 and will continue in the first instance for a year.

The CAT2 system has two basic parts, the CAT2 formalism and its implementation (the software) and the CAT2 lexica, grammars and translation modules (the lingware). In the following section we want first to give a minimal overview of the basic architecture of the CAT2 MT system. Secondly, we describe some aspects of the linguistic base component, which allows for the treatment of the syntactic and phraseological phenomena of real-life text without losing its internal intelligibility and thus the possibility of being easily maintained and ameliorated.

2.1 Architecture of CAT2

The basic architecture of the CAT2 system is that of a classic stratificational, transfer-based one, using tree structures at all levels. In figure 1 MS refers to 'morphological structure', CS to 'constituent structure' and IS to 'interface structure'. T1 and T2 are intermediate levels.

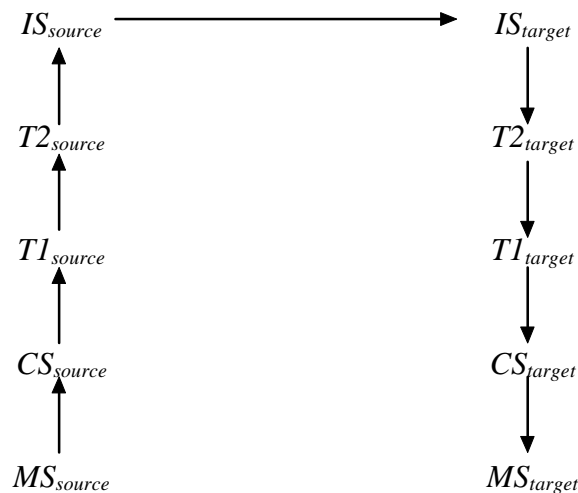


Fig. 1: Translation Path of the CAT2 System

Following current sign-based approaches in linguistics (cf. [3] Pollard & Sag 87), syntactic and semantic analysis is done simultaneously at the constituent level (CS). By the joint cooperation of syntax and semantics, the creation of spurious analyses due to structural ambiguities can be reduced. The levels T1 and T2 are intermediate levels which serve the purpose of dividing up the stages of analysis/synthesis between CS and IS into more manageable tasks. The IS level, having no linguistic justification other than that of serving for transfer, can be shaped and restricted according to these demands. In particular, the IS is designed to allow compositional translation of each of the major constituents, relegating the task of verifying the resulting target structure to the target CS generator. This makes the IS structure more efficient for transfer and more intelligible for the linguist/translator.

The CS level and the IS level are connected by levels T1 and T2 and between these four levels a number of general mapping rules elide in analysis functional categories (determiners, argument prepositions, complementizers, auxiliaries etc...), undo every kind of "movement" (e.g. passivization, extraposition, etc.), and replace the syntactic predicates which are semantically empty (so-called light verbs and copulas) by the corresponding semantic predicate (the predicative noun or adjective), as well as introducing 'dummy' pronouns for control verbs, for example – to represent missing subjects and objects. In generation these mapping rules perform the reverse, i.e. generate functional categories, move elements to surface position, and try to introduce 'dummy' predicates wherever possible.

This symmetrical architecture allows for the reversibility of the system. Transfer has to be underspecified as regards surface features (word order, choice of articles and prepositions as well as category, of morpho-syntactic voice, tense and aspect marking) if the transfer component is not to resemble that of a commercial kick-and-rush system. The abstracting away from surface features and the reliance on semantic and pragmatic aspects in transfer (attained by the inclusion of general cognitive categories relating to time, space and cause in the IS representation), is an approach located somewhere between a normal word-based transfer and an interlingua, and presupposes a fully competent language component which can relate the semantic and pragmatic content to its surface representations.

2.2 Development of the Linguistic Core Component

2.2.1 Use of common modules Often the same linguistic information must be available in different places of the NLP system – e.g. grammatical rules for testing the completeness of an argument structure, or common lexical entries (terminology, punctuation marks, etc...) may be used by different language components. CAT2 allows grammatical resources to be declared as a COMMON module. These common modules can then be called by any other module (i.e. language-specific module) with the CALL statement, just like a subprocedure or function in most ordinary programming languages.

The advantages of this solution are obvious: from a linguistic point of view, rules need be written and modified only once, which augments the consistency of the grammatical information. In addition, it allows for a parametrized approach to grammar, where the language-specific grammar is composed of universal rules, parameters of variation and language-specific rules, following the ideas developed by Chomsky [1] (Chomsky 81).

As a consequence grammars of new language components are written in a very short time by setting the parameters appropriate to the new language. These common modules have, of course, speeded up the process of introducing Arabic greatly.

To give an example of common rules, we describe the three main types of building rules which are used in the English, French, German, and (now) Arabic, syntactic components. These are responsible for the detection of arguments (i.e predicate–argument structures), the detection of modifier relations (i.e. head–modifier structures) and the subcategorization by functional categories (functional head–argument structure).

Predicate–Argument Structures

Arguments to predicates are specified in the frame feature of each lexical entry. Two general–purpose structure building rules (called "b–rules") are responsible for the detection of argument relations during parsing. Since the b–rules in the CAT2 formalism not only specify dominance relations but also linear order of the constituents, the two b–rules necessary to express the predicate–argument relation are (a) when the argument precedes the predicate (e.g. in German), and (b) when it follows the predicate (e.g. in English). A simplified version of the b–rule responsible for argument detection to the right is reproduced here:

```
{head=HEAD} . [ {head=HEAD, frame= ( {arg1=ARG};
                                     {arg2=ARG}; {arg3=ARG}; {arg4=ARG} ) } , ARG ] .
```

The rule is read as follows: A lexical category which is the head of a projection ({head=HEAD}) and hereby sharing all head features with the mother node takes a constituent to its right as an argument if the constituent unifies with one of the descriptions in ARG1, ARG2, ARG3 or ARG4.

Head–Modifier Structures

A constituent is analyzed as a modifier of another constituent if the restrictions ({restr=RESTR}) imposed by the modifier unify with the description of the constituent being modified. Again, two rules accommodate modifiers to the left of the head (eg in adjectives in English) and modifiers to the right (eg in adjectives in Arabic); an example of the former is shown here:

```
{head=HEAD} . [ {role=mod, head={restr=RESTR} } , {head=HEAD}&RESTR ] .
```

Functional Categories

In our implementation, we also assume that the functional categories form head projections. Following Grimshaw's **Extended Projections** [2] (Grimshaw 91), nouns combine with determiners to form extended noun projections, and prepositions combine with extended noun projections to form larger extended noun projections. In the same way, complementizers combine with verbs to form extended verb projections. All functional categories except coordinators have one argument position, coordinators have two. Here is the rule responsible for functional categories in initial position:

```
{head=HEAD} . [ {role=funct, head=HEAD, frame= ( {arg1=ARG}; {arg2=ARG} ) } , ARG ] .
```

2.2.2 Simplification of the grammar: The use of extended head features The features that relate to the entire extended projection are contained in the ehead (extended head) feature. For example, if a noun phrase contains the +wh feature (e.g. "which country"),

then a prepositional phrase containing this noun phrase also contains the *+wh* feature (e.g. "in which country"). By definition, *eh* features are a subset of head features.

Although head features and subcategorization features are regarded as standard in linguistic circles, extended head features are not yet part of mainstream linguistics. This is astonishing insofar as the extended head feature principle as formulated by Grimshaw [2] (Grimshaw 91) seems to be a fundamental principle of natural languages. First of all, there exists an undeniable difference between head features and extended head features. *VFORM* (verb form, e.g. finite, nonfinite, participle, etc.) is a typical head feature which is available at the maximal VP projection of the verbal head. But this information is not passed up higher than the local VP, since higher VP projections may have different *VFORM*s (e.g. when an auxiliary with one *VFORM* takes a VP projection having a different *VFORM* as an argument, forming an extended VP projection). Information about tense and aspect, on the other hand, must be present at every point in an extended VP projection, passing the morpho-syntactic realization of tense and aspect up to S, where the adjunction of temporal modifiers must be controlled and where the tense-aspect information must agree with possible complementizers. For a more detailed description of extended head features and their function within an MT system see [4] (Sharp & Streiter 92).

Several problems have arisen in this respect during the writing of the Arabic grammar. An example is that of modal constructions. Modal verbs are in fact functional words – ie they do not have independent meaning, but only 'modify' the meaning of a particular head projection (in this case that of the main verb). They are therefore represented at the abstract level (IS) as features of the main verb, and not as nodes themselves. As a consequence the structures they are part of should be built with the building rule for functional categories given above. We show it again with more detail.

```
{head=HEAD} . [ {role=funct , head=HEAD&{ehhead=EH} ,
                 frame=( {arg1=ARG&{head={ehhead=EH}}
                          ; {arg2=ARG&{head={ehhead=EH}} } ) } ,
                 ARG ] .
```

In Arabic the main verb of the modal sentence is introduced by the complementizer *إن* which puts it in subjunctive mode. The mode feature has always been considered to be an extended head feature, as in the languages treated thus far in Cat2 the mode within a VP never varied within its extended head projection.

The application of the above rule to create the correct structure passes this information first to the mother node dominating the complementizer and the main verb and second to the modal verb. This last value is obviously not correct (the correct sentence is given in the following example).

e.g. يستطيع أن يخزن الملفات

can_{pres,sing,masc,ind} store_{masc,sing,subj} the-files

He can store the files.

To avoid this error we will use a head feature *mode* which in this sentence takes the value *ind* and which concerns only the modal verb. The generation of the morphological form of this verb should take into account this feature and not the extended head one.

- D AGENT MALE (wazn=2) مدرس (mudarris - teacher (male))
- D AGENT FEMALE (wazn=2) مدرسه (mudarrisa - teacher (female))
- D PATIENT (wazn=2) مدرس (mudarras - taught material)

The subcategorisation frame is described in the 'frame feature' using the CAT2 system of roles and selectional restrictions, and is the same for each derivation. The coindexing of derivational forms with a particular argument position is linked via the 'refindex' feature (e.g. مدرس (mudarris - teacher)) fills the first argument position, which prevents it from being filled by another element if this derivational form is present in the sentence).

The derivational richness in Arabic, however, which ideally can only be advantageous in this form of lexicon-writing as it enables one stem and its derivations to be more quickly and consistently coded, has in fact proved to be a problem. During the coding of our main languages in Cat2 (German, English and French) we found it easy to include different derivations of a lexeme in one entry, as these are generally limited to either merely variations in syntactic category of the lexeme (ie mean the same as the lexeme they are derived from), or variations in the slot the derivation fills in the subcategorisation frame. In both these instances the derivational forms share the same subcategorisation frame, and can therefore be described in one lexical entry. For example the word 'sell' in English has the following derivations: 'selling' (noun), 'sale' and 'seller'. The first two are nominal forms of the verb, which, along with the verb, fill the zero argument slot (ie the process itself), and the last is the agentive derivation, ie the first argument of the process.

In Arabic, however, there are two different sorts of derivations – direct derivations from the root, and derivations with different 'weights' (the 'additional verbs' and their derivational forms). The first type only creates a problem when the derivation does not fill a slot in the frame. This is the case with words such as مدرسه (madrasa) (school) – ie a place where one studies. This derivation in a sense provides insight into additional frame information and can be seen as a locative argument to the verb, and it is debatable whether this should be given a slot in the subcategorisation frame (for why should the locative be any less an argument than the subject or object?), or whether it should be seen as a modifier. We decided that it should remain a modifier because locatives are generally governed by other predicates (eg 'in the school'). We were able to enter these in the same entry for درس (darasa), even though it doesn't have a subcategorisation frame nor a place in the frame of other derivations in the entry by writing rules which do not allow locative derivations to use the frame – ie forbidding them to subcategorise for any other item in the sentence.

The other type of derivation proved more problematic, however, as here the deviations from the root form (ie the lexeme) are both syntactic and semantic. the lexeme خرج (kharaja), for example, has the following 'weights': خرج (kharaja) (one); خرج (kharraja) (two); تخرج (takharraja) (five). We found that it was relatively easy to include the 2nd form of the verb in the entry for the root form, as where both forms exist it is almost always the transitive form of the root (here: خرج (kharaja) – 'to exit', خرج (kharraja) – 'to make (s.o.) exit'), and for this it is easy to give the entry a frame description of:

```
{ frame={ arg1={ AGENT } ,
          arg2={ THEME } } }
```

and write default rules saying that the first (root) form cannot subcategorise for the first argument – ie has no agent. Someone who exits has the role 'theme', whether of their own accord, or forced. A similar procedure is possible for, for example, the 7th form, which is the passive form (equivalent to the ergative), and also the 8th form, which is the reciprocal form.

For the fifth form derivation, however, which means 'to graduate' (ie 'to exit from university successfully'), because it deviates not only syntactically but also semantically from the root form, it is impossible at the present stage of development of Cat2 to exploit the 'derivedness' and incorporate this form in the same entry as the others – another frame, another entry. For the moment we have to be satisfied with just taking advantage of the derivations based on syntactic deviation rather than semantic deviation.

3 THE MORPHOLOGICAL GENERATOR NALG (NATURAL ARABIC LANGUAGE GENERATION)

The NALG morphological generator has been developed at the ERI since 1984. It is a system which works on the micro level of generation – i.e. the word level. It takes Arabic roots and generates different word forms according to Arabic morphology rules. It functions using pattern–matching methods. Figure. 2 shows a simplified block diagram of NALG.

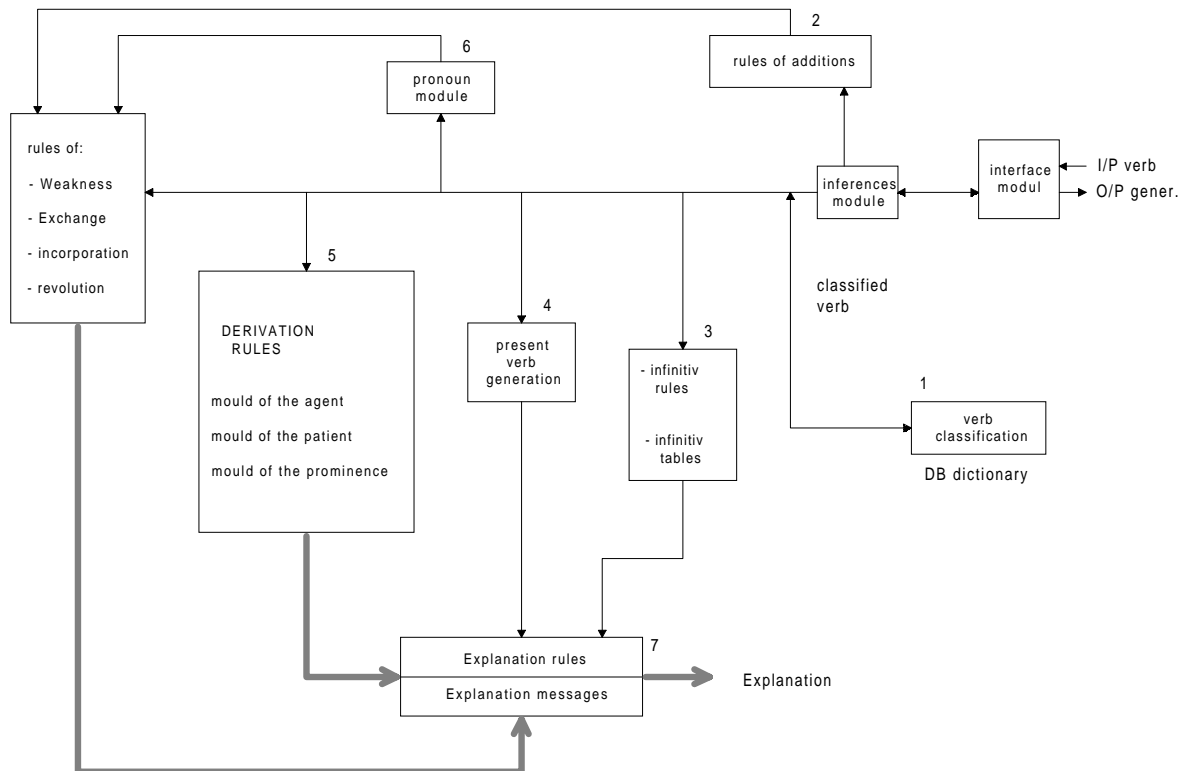


Fig. 2: The NALG Morphological Generator

3.1 NALG Architecture

The NALG system comprises three main components:

- a knowledge base, which consists of a set of facts and rules about Arabic morphological patterns;
- an inference engine, which processes the knowledge base and generates different words using a set of rules (in other words which controls the activity of the NALG process);
- an explanatory interface, which illustrates how rules are applied and intermediate results are obtained.

The Knowledge Base

Knowledge is stored in the dictionary in the form of objects. Each object is stored with its own attributes. These attributes are made up of the root and various characteristics which affect what derivational forms it has – for example which weights the root can have, whether it contains weak letters (which indicate changes in inflectional and derivational forms), transitivity (which tells us whether the root has a 'patient' derivational form or not), etc. An example of an object is given in figure. 3.

Object	Attributes				
abstract	verb class	verb	infinitive	transition	derivation
past tense		construction	weight for		
verb			abstract		

Example:

كتب	نصر	صحيح	كتابه	متعدى	متصرف
-----	-----	------	-------	-------	-------

Fig. 3: A lexical object in NALG

The Inference Module

The inference engine contains a set of production rules which can be represented in the form of:

IF condition THEN action

During the execution, once the condition is satisfied, the production or the corresponding action can take place. The elementary knowledge of the NALG can be put in the form of production rules. The interpreter has the task of deciding what to do next in order to generate the correct forms of a word. This can be done by applying an appropriate sequence of rules to an initial task domain situation. In the NALG algorithm different rule sequences are considered, and the representation is guided by the data structures.

Explanatory Module

The explanatory interface assists the user – when required – with explanation messages that explain how and why a certain output has been given. The explanation tractor keeps track of all the constraints that are present and all the rules that are fired during the generation process, and then assigns suitable explanation messages.

3.2 Capabilities of NALG

The NALG system can, from a root, generate the different weights of the root (i.e. the different additional verbs from the abstract verb) and the various derivational forms of this weight. NALG can generate 12 different weights, and can, as well as being able to generate inflectional endings, plural forms, different tenses of a verb, etc, it can also perform the following derivations from various weights:

- abstract noun
- mould of agent
- mould of patient
- adjective mould used as agent
- mould of prominence
- mould of exaggeration
- noun of one action
- noun of kind
- noun of "mimi"
- artificial noun
- noun of instrument
- noun of time
- noun of place

4 COMBINING NALG AND CAT2

Both the NALG and the CAT2 systems are based on deep lexical analysis, though for different reasons. The CAT2 system bases its lexical representation on the reduction of different lemmata to their original lexeme as it bases translation on semantics, where the syntactic structure of a word/ sentence in the source language may have little bearing on the syntactic structure of that of the target language. This is made possible by transferring the semantics of the source language and creating a large choice for possible ways of realising this in the target language. In NALG, the descriptions of words are based on the same derivational information, though this time for morphological reasons – because Arabic is an extremely patterned language, and any morphological description of the language does well to exploit this fact. This common form of analysis made the task of combining the two programs relatively easy, and meant that the common lexicon could be written based on the analysis of the Arabic derivational morphology used in NALG and used for both the syntactic and morphological parts of the generation process.

5 TRANSLATING MEDICAL TERMINOLOGY AND MEDICAL TEXTS

The medical field in the Arab countries is dominated by the English and French languages. In most countries the language used in universities for the study of medicine

is either English or French, and this phenomenon is carried over to the practice of medicine, where medical diagnoses, progress reports, prescriptions etc are all written in the appropriate foreign language. One of the basic assumptions of the CEC is that a person should have the right to speak and work in his/her own language. Although the ambition to change this situation in the medical field is obviously unrealistic for this project, it hopes to achieve a step in this direction. The medical text type chosen as a corpus for Aramed is the information slip accompanying medicines with instructions for use, ingredients of the drug, what symptoms it treats, its dosage and possible side effects, etc. The project deals with only instructions for use, in order to keep the sublanguage relatively small. Most medical texts in the Arab world, written in a foreign language, never appear in Arabic. This area, however, is a useful area to work in, as these information slips are indeed written in English in many Arab countries by specialists (doctors, chemists and so on) and translated into Arabic for the home market. Both the original and the translation are included with the medicines – a fact which makes it easy to write translations for medical terminology and to check the results of the translation as a translation is already available.

As mentioned above, the CAT2 system has already worked in the field of medicine in the ANTHEM project, and thus has already been used for the translation of medical texts. Part of the methodology (i.e. the idea of using terminological codes as a type of interligua) employed here can to a certain extent be used for Aramed. In Anthem only noun phrases are translated, the whole of which can be analysed with one or more SNOMED codes. In Aramed, however, whole sentences are translated, and not just NPs, and these sentences contain units without SNOMED codes. These are therefore represented as normal IS objects and go through the normal transfer process in order to be translated to another language.

6 CONCLUSION AND FURTHER DEVELOPMENTS

Aramed has successfully included an 'exotic' language in a MT system written primarily for (a few) European languages, and has incorporated in the MT system a morphological generator which, although developed separately, exploits the same derivational morphology as the MT system. This has been achieved by virtue of the fact that the system is structured with a high degree of modularity, and its treatment of linguistic concepts is as universal as possible, enabling the similarities between languages, which are numerous even between those as different as English, German and Arabic, to be exploited.

The Aramed project is a relatively small project in terms of manpower and duration, and it is hoped that the work being done can be extended after the initial project is finished. The extent of the field it has begun to deal with is huge; the SNOMED codes number 130,000, and there are many more text types in the field of medicine, containing many different types of sentence structure than the area dealt with in Aramed. We think that Aramed contributes an important drop for Arabic in this medical ocean.

References

- [1] Noam Chomsky. *Lectures on Government and Binding*. The Pisa Lectures. Studies in Generative Grammar 9. Foris Publication, Dordrecht Holland & Cinnaminson U.S.A., 1981.

- [2] Jane Grimshaw. *Extended Projection*. Brandeis University, Waltham MA 02254, ms, July 1991.
- [3] Carl Pollard and Ivan Sag. *Information-Based Syntax and Semantics*. CSLI Lecture Notes 13, Stanford, 1987.
- [4] Randall Sharp and Oliver Streiter. *Simplifying the Complexity of Machine Translation*. Meta-92, pages 681-692, 1992.