

**Workshop on  
Terminology design: quality criteria and evaluation methods  
(TermEval)**

Magazzini del Cotone Conference Centre – Genoa – Italy  
In Association with the

5<sup>th</sup> International Conference on Language Resources and Evaluation  
LREC 2006 <http://www.lrec-conf.org/lrec2006>  
28<sup>th</sup> May 2006

09:00 – 09:30 **Opening by Chair**

**Session 1**

09:30 – 10:10 **i-Term for Nordterm**

Bodil Nistrup Madsen, Hanne Erdmann Thomsen, Annemette Wenzel

10:10 – 10:50 **Engaging Customers in Terminology Creation For Windows Vista**

Barbara Karsch

10:50 – 11:00 Discussion

11:00 – 11:30 *Coffee Break*

**Session 2**

11:30 – 12:05 **Some note about the evaluation of terms extraction systems**

Jorge Vivaldi, Horacio Rodríguez

12:05 – 12:40 **An application-oriented terminology evaluation: the case of back-of-the book indexes**

Touria Aït El Mekki, Adeline Nazarenko

12:40 – 12:35 **Semi-automatic Checking of Terminographic Definitions**

Selja Seppälä

**13:30 – 14:30** *Lunch Break*

**Round table: Evaluating terminological work: needs and methods**

14:30 – 14:40 **Introduction**

14:40 – 15:10 **A point of view from computational terminology**

15:10 – 15:40 **A terminologist point of view**

15:40 – 16:20 **General discussion**

# Workshop Organiser

## Organisation Committee

Rute Costa, Centro de Linguística da Universidade Nova de Lisboa, Portugal  
Fidélia Ibekwe-SanJuan, Unité de Recherche en Science de L'information et du  
DOCument, France  
Susanne Lervad, Danish Center for Terminology, Denmark  
Marie-Claude L'Homme, Observatoire de Linguistique Sens-Texte, Canada  
Adeline Nazarenko, Laboratoire d'Informatique de Paris-Nord, France  
Henrik Nilsson, Terminologiecentrum, Sweden

## Programm Committee

Olivier Bodenreider, NLM, USA  
Antia Basse, University of Maiduguri, Nigeria  
Gerahrd Budin, Universität Wien, Austria  
Teresa Cabré, University Pompeu Fabra, Spain  
Gorgeta Ciabanu, Universitatea Politehnica, Romania  
Anne Condamines, ERSS, France  
Rute Costa, Centro de Linguística da Universidade Nova de Lisboa, Portugal  
Claude de Loupy, France  
Valérie DeLavigne, DYALANG, France  
Patrick Drouin, Université de Montréal, Canada  
Fidelia Ibekwe-SanJuan, Université de Lyon 3, France  
Kyo Kageura, Graduate School of Education, University of Tokyo, Japan  
Marie-Claude L'Homme, Observatoire de Linguistique Sens-Texte, Canada  
Susanne Lervad, Danish Center for Terminology, Denmark  
Teresa Lino, Centro de Linguística da Universidade Nova de Lisboa, Portugal  
Marie-Pierre Mayar, CTB, Belgium  
Adeline Nazarenko, Laboratoire d'Informatique de Paris-Nord, France  
Fidelma Ni Ghallchobhair, NSAI, Ireland  
Henrik Nilsson, Terminologiecentrum, Sweden  
Anita Nuopponen, University of Vaasa, Finland  
Maria Pozzi, El Colegio de México, México  
Martin Rajman, EPFL, Switzerland  
Raquel Silva, Centro de Linguística da Universidade Nova de Lisboa, Portugal  
Arvi Tavast, ETER, Estónia  
Sue Allen Wright, ANSI, USA

# Table of Contents

<b>i-Term for Nordterm .....</b>	<b>1</b>
Bodil Nistrup Madsen, Hanne Erdmann Thomsen, Annemette Wenzel	
<b>Engaging Customers in Terminology Creation For Windows Vista .....</b>	<b>8</b>
Barbara Karsch	
<b>Some note about the evaluation of terms extraction systems .....</b>	<b>12</b>
Jorge Vivaldi and Horacio Rodríguez	
<b>An application-oriented terminology evaluation: the case of back-of-the book indexes.....</b>	<b>18</b>
Touria Aït El Mekki, Adeline Nazarenko	
<b>Semi-automatic Checking of Terminographic Definitions.....</b>	<b>22</b>
Selja Seppälä	

# Author Index

Adeline Nazarenko  
Annemette Wenzel  
Barbara Inge Karsch  
Bodil Nistrup Madsen  
Hanne Erdman Thomsen  
Horacio Rodríguez  
Jorge Vivaldi  
Selja Seppälä  
Touria Ait El Mekki

# i-Term for NORDTERM

Bodil Nistrup Madsen<sup>\*</sup>, Hanne Erdman Thomsen<sup>†</sup>, Annemette Wenzel<sup>\*</sup>

<sup>\*</sup>DANTERMcentret / <sup>†</sup>Department of Computational Linguistics  
Copenhagen Business School, Dalgas Have 15, DK-2000 Frederiksberg, Denmark  
bnm.danterm@cbs.dk, het.id@cbs.dk, aw.danterm@cbs.dk

## Abstract

In this paper we will present the use of the Internet terminology and knowledge management system, i-Term, for the NORDTERM project: *Terminology of terminology in Nordic languages*. We will describe how the data from the original document-based vocabulary is extracted and imported into the i-Term database. We also present how i-Model is used to construct concept systems, and how the information inherent in the concept systems is automatically integrated into the termbase entries. Finally we will give some examples of problems encountered in this multilingual terminology project.

## 1. Introduction

The goal of the NORDTERM project: *Terminology of terminology in Nordic languages* was to establish a vocabulary comprising terms, definitions, notes and concept systems in Danish, Faroese, Finnish, Greenlandic, Icelandic, Norwegian (Bokmål and Nynorsk), Sami and Swedish within the domain of terminology.

As input for this project served the vocabulary *NORDTERM 2, Terminologins terminologi* (Terminology of Terminology in Nordic) from 1989 as well as *ISO 1087-1:2000 (1087-1) Terminology work -- Vocabulary -- Part 1: Theory and application*. During 2004 and 2005 the original vocabulary was revised and extended. All NORDTERM partners participated in this work, which was coordinated by Terminologisentrum TNC (Swedish Centre for Terminology), Sweden, cf. the website of NORDTERM, <http://www.nordterm.net>.

Since the project partners are geographically scattered, there was a need for a tool for optimising co-operation. It was required that the system should be web-based, so that information entered by the local groups would be immediately available to the partners in the other countries. In order to make the working process dynamic, it was also a requirement that questions and comments could be entered directly into the database. In this way there would be no need for circulation of reviewed versions of the vocabulary, and problems of version management could be avoided. The project group also wanted a way of publishing concept systems together with the other types of information.

The system chosen was i-Term, developed by the Danish Terminology Centre, DANTERMcentret, <http://www.danterm.dk>. This system meets the requirements specified by the NORDTERM project group, and furthermore i-Term has a graphical module, i-Model, which allows the user to create concept systems comprising all kinds of relations between concepts, characteristics of concepts and subdivision criteria, as recommended in the ISO standards for terminology work.

Below we will describe the process from vocabulary to database, and the use of i-Term and i-Model as a tool for elaboration, revision and publishing of information about concepts, including concept systems. At the same time we will give some examples of the terminological methods and problems that may arise in multilingual terminology work.

## 2. From vocabulary to database

The first step was to convert the vocabulary data. The import function of i-Term allows the user to import data into the database from an XML file.

In Figure 1, an entry from the original vocabulary in the Nordic Terminological Record Format, NTRF, is presented.

```
enTE appellation
enSYTE name
enDF verbal designation of an individual concept
      [ISO 1087-1:2000]
frTE appellation
frSYTE nom
svTE egennamn
svSYTE proprium
svDF benämning på ett individualbegrepp
svAN Egennamn skrivs normalt med stor begynnelse-
      bokstav. Exempel: Jupiter, Östersjön.
BD 6 Benämningar
```

Figure 1: Example of data from the vocabulary Terminology of terminology in Nordic languages

The MS Word file was converted automatically into an XML file, which could then be directly imported into the database.

```
<article id="5">
  <subject code="1.0.0">
    Nordterm</subject>
  <concept id="13">
    <language code="en">English
    </language>
    <gndef>verbal [[designation]] of
    an individual concept</gndef>
    <gndefref>ISO 1087 1:2000
    </gndefref>
    <comment></comment>
    <term>
      <value>appellation</value>
    </term>
    <term>
      <value>name</value>
    </term>
  </concept>
```

```

<concept id="14">
  <language code="se">Swedish
  </language>
  <gendef>[[benämning]] på ett
  [[individualbegrepp]]</gendef>
  <gendefref>ISO 1087 1:2000
  </gendefref>
  <comment>Egennamn skrivs normalt
  med stor begynnelsebokstav.
  Exempel: Jupiter, Östersjön.
  </comment>
  <term>
    <value>egennamn</value>
  </term>
  <term>
    <value>proprium</value>
  </term>
</concept>
</article>
<system id="6 Benämningar">
</system>

```

Figure 2: Example data from Figure 1 in XML

Figure 2 shows the data from Figure 1 with XML markup (French term omitted).

Figure 3 shows the same data as they are presented in i-Term, after the introduction of information on concept relations concerning the Swedish concept. This information is included in the entry by i-Model when the user draws the concept system as described in section 3.

After import of the original data in English, French and Swedish, the project partners in the other Nordic countries were able to add their language-specific data directly into the database. This made the next stages of work much easier. First, the project coordinator did not have to copy the data into a document in the Nordic Terminological Record Format, NTRF. Second, proof reading and subsequent corrections could be left to the native speakers of each language.

Figure 3: Example data from Figure 1 and 2 in an i-Term article.

### 3. Establishment of concept systems

The vocabulary already comprised concept systems in Swedish (produced as drawings in PowerPoint). These concept systems had to be reconstructed manually in i-Model, as there is no way to go from the graphics of PowerPoint to the structured data of i-Model. Figure 4 shows the concept system *Designations* in the original format with English glosses

In Figure 5 the corresponding Danish concept system is presented in i-Model (English translation of the Danish system). In this case the Danish concept system differs slightly from the original Swedish concept system due to different definitions of the concepts in the two languages. This is discussed in more detail in section 4.

At this point we would like to draw attention to the fact that the concepts (in grey boxes) are related directly with the respective entries in the i-Term database. This means that all information is accessible both from the termbase view shown in Figure 3 and from the concept system view in Figure 5. In other terminology management systems graphic representations of concept systems may be included as static graphical objects. In i-Model, the concept systems may be changed dynamically.

Another feature worth mentioning here is the possibility to add criteria of subdivision (the white boxes).

The feature specifications representing the characteristics of the concepts (under the grey concept boxes) will be presented in more detail below.

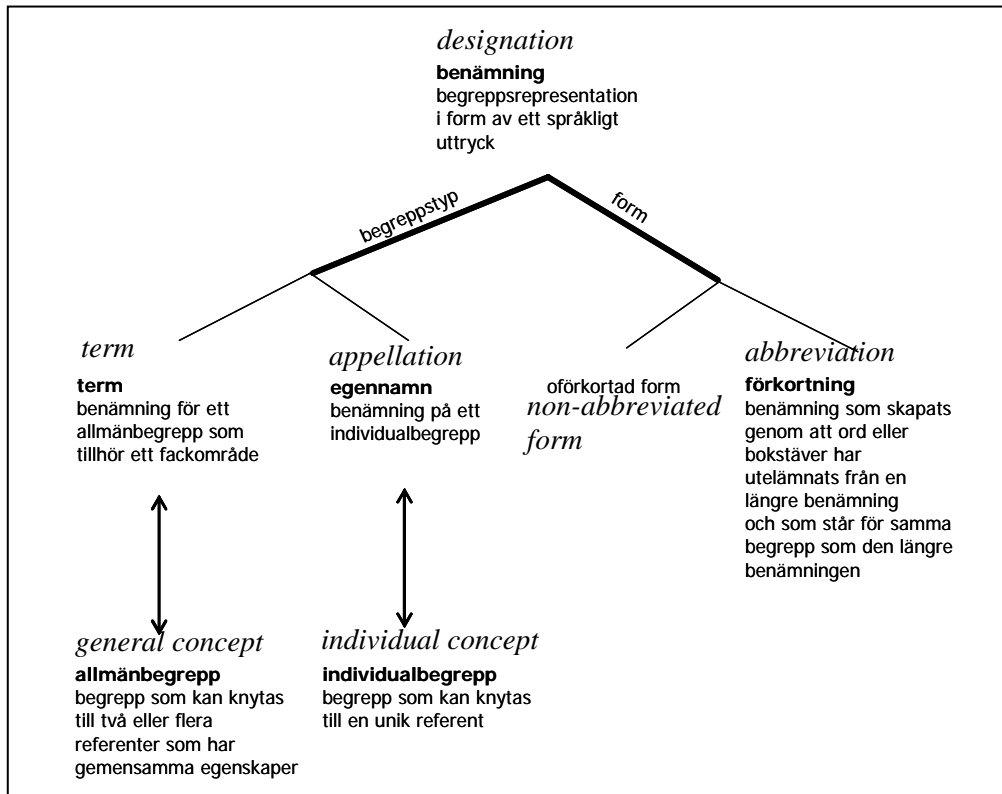


Figure 4: Example of a Swedish concept system (Designations) from the vocabulary

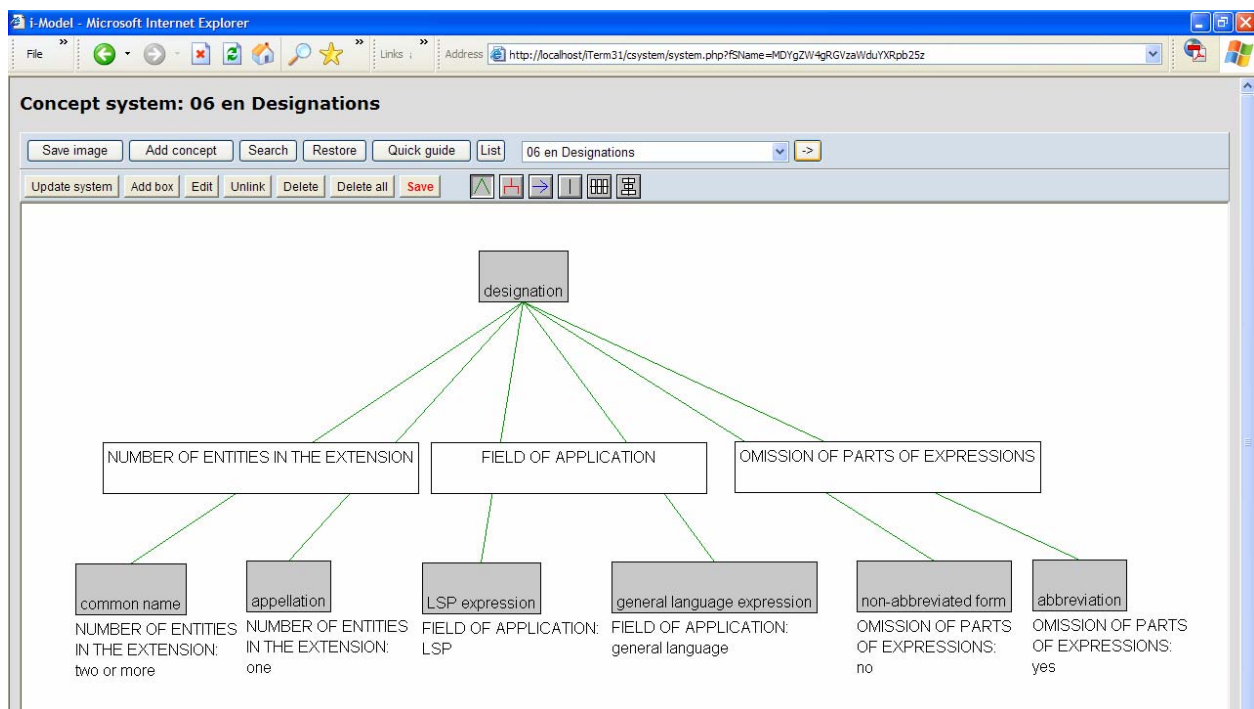


Figure 5: The Danish concept system (Designations) in i-Model (translated into English)

In the next figures we illustrate the process of establishing this concept system in i-Model.

First all concepts that have already been marked as belonging to this concept system, are retrieved in i-Term. The result is shown in Figure 6. All concepts are

added to the concept system *06 Designations* by means of the button: *Add to system* and the concepts are ready to be related by means of concept relations as shown in Figure 7.

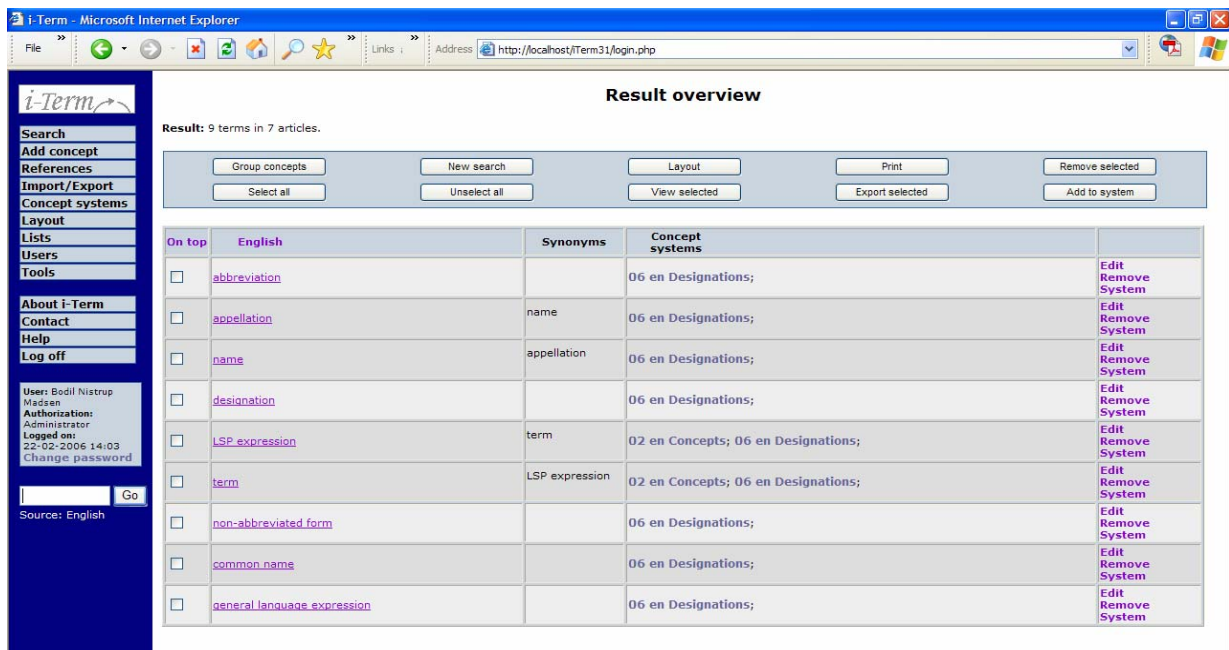


Figure 6: Concepts belonging to concept system 06 en Designations

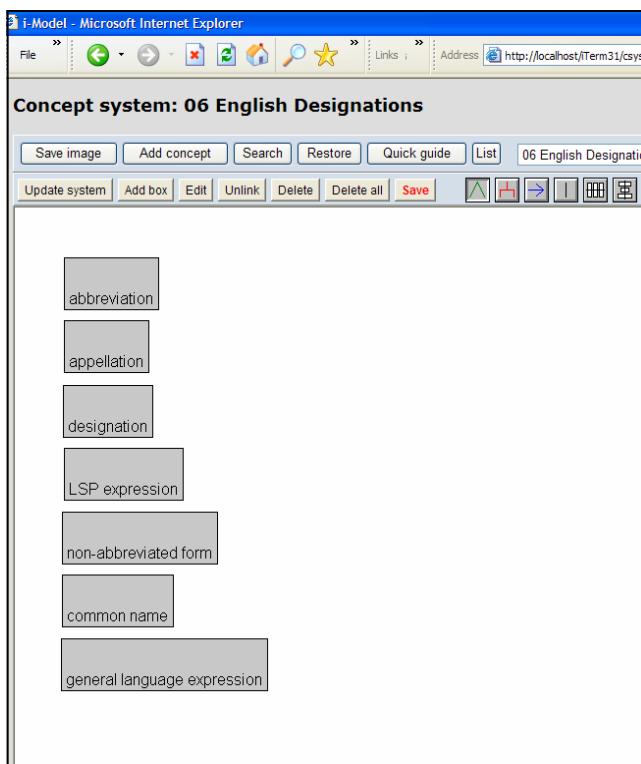


Figure 7: Concepts from Figure 6 before introduction of concept relations

Figure 7 illustrates the starting 'mode' for any concept system in i-Model: all concepts included are shown with no relations. The concept boxes can be dragged to other positions in the window, and relations are drawn between them. I-Model allows the user to select from three basic relation types and any associative relations. The choice is made on the panel above the window: the generic relation (green lines), the partitive relation (red lines) and the temporal relation (blue arrow). Other associative relations will be drawn as black lines, and the user may choose to

add arrows on the lines and a relation name, which will appear in the diagram. In the diagrams in Figures 9 and 12 we have added arrows to the associative relations to distinguish them from the generic relations, as the printing here does not render the colors.

In Figure 8, three concepts have been related by means of the generic relation. As soon as the concept system is saved, the information about these relations will be accessible in the termbase, as illustrated also in Figure 3.

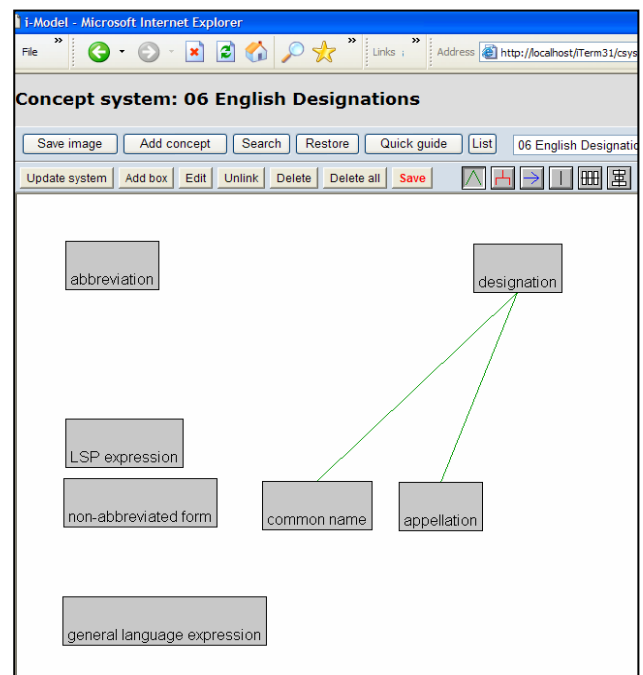


Figure 8: Two concept relations introduced

Figure 9 gives the Danish concept system for types of concepts (Danish system translated into English).



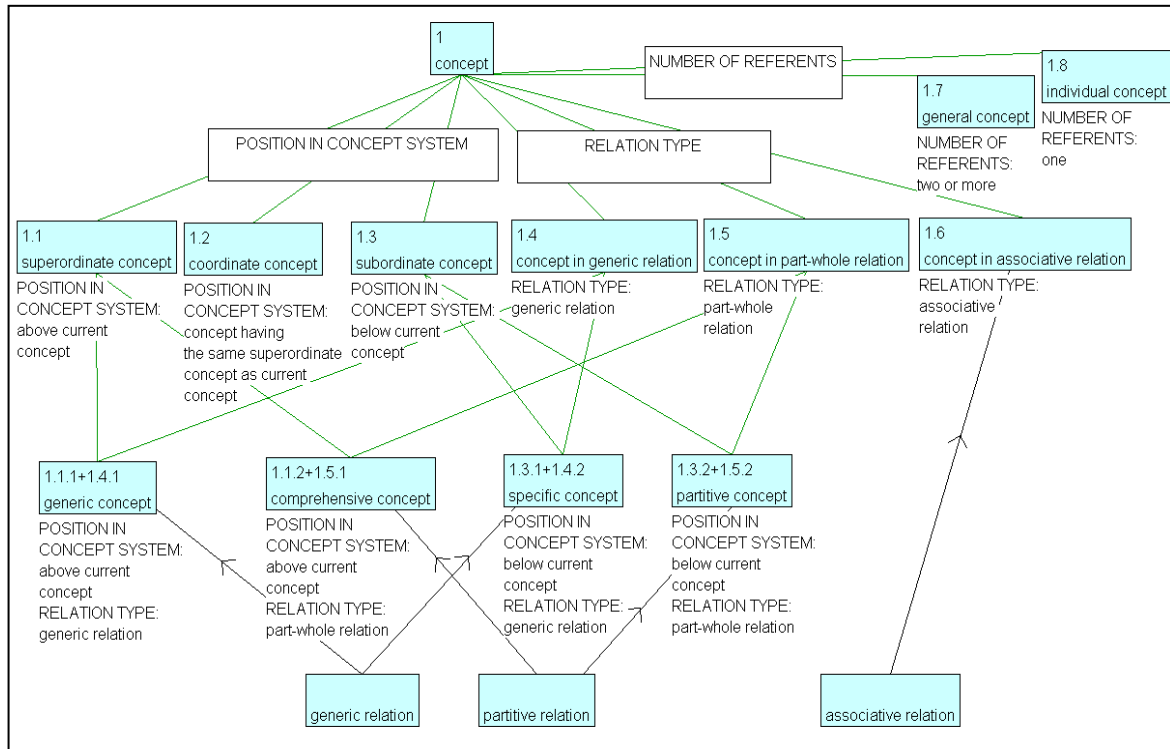


Figure 9: Danish concept system for concept types

In this concept system characteristics and subdivision criteria have been introduced. The principles of using feature specifications to model characteristics of concepts have been developed in the CAOS project, cf. (Madsen, Thomsen and Vikner 2005).

In the concept system in Figure 9 we have furthermore introduced systematic notations. This allows the user to produce a systematic list like the one in Figure 10. If desired, the definitions could also be presented in this list.

Concept	Notation	Characteristic feature
concept	1	
superordinate concept	1.1	POSITION IN CONCEPT SYSTEM: above current concept
generic concept	1.1.1+1.4.1	POSITION IN CONCEPT SYSTEM: above current concept RELATION TYPE: generic relation
comprehensive concept	1.1.2+1.5.1	POSITION IN CONCEPT SYSTEM: above current concept RELATION TYPE: part-whole relation
coordinate concept	1.2	POSITION IN CONCEPT SYSTEM: concept having the same superordinate concept as current concept

Figure 10: Part of a systematic list

Figure 11 illustrates how the terminologist may edit characteristics and definitions simultaneously with the construction of the concept system. In this case the concept *LSP expression* has been chosen for edition, and the small edit window pops up.

#### 4. Special problems in multilingual terminology work

When a group of terminologists from different countries work together, they will sometimes have different understandings of concepts and different traditions for structuring the domain. In such cases i-Model is a good tool for making the differences between the concepts clear, as they can be illustrated in different concept systems and feature specifications for the languages involved.

The NORDTERM project described here is descriptive and therefore we have allowed such differences to be maintained in the termbase. In a harmonization project, the result in the NORDTERM termbase would form a first stage where differences are discovered. In later stages it would be decided how to harmonize and which compromises would have to be made. Such a harmonization project would also benefit from the clarification made possible in i-Model.

In the remaining part of this section, we will focus on some of the different views on the terminology of terminology that we have discovered in the NORDTERM project.

The Danish concept system in Figure 5 comprises the concepts *common name* and *general language expression*, which were not in the Swedish system in Figure 4. These were introduced in order to allow for a differentiation between general language expressions and expressions of LSP, a distinction which is not clarified in the Swedish system. The concepts *common name* and *appellation* in the Danish concept system differ with respect to number of entities in the extension and may be found both in general and specialized language. The concepts *general concept* and *individual concept* related to *term* and

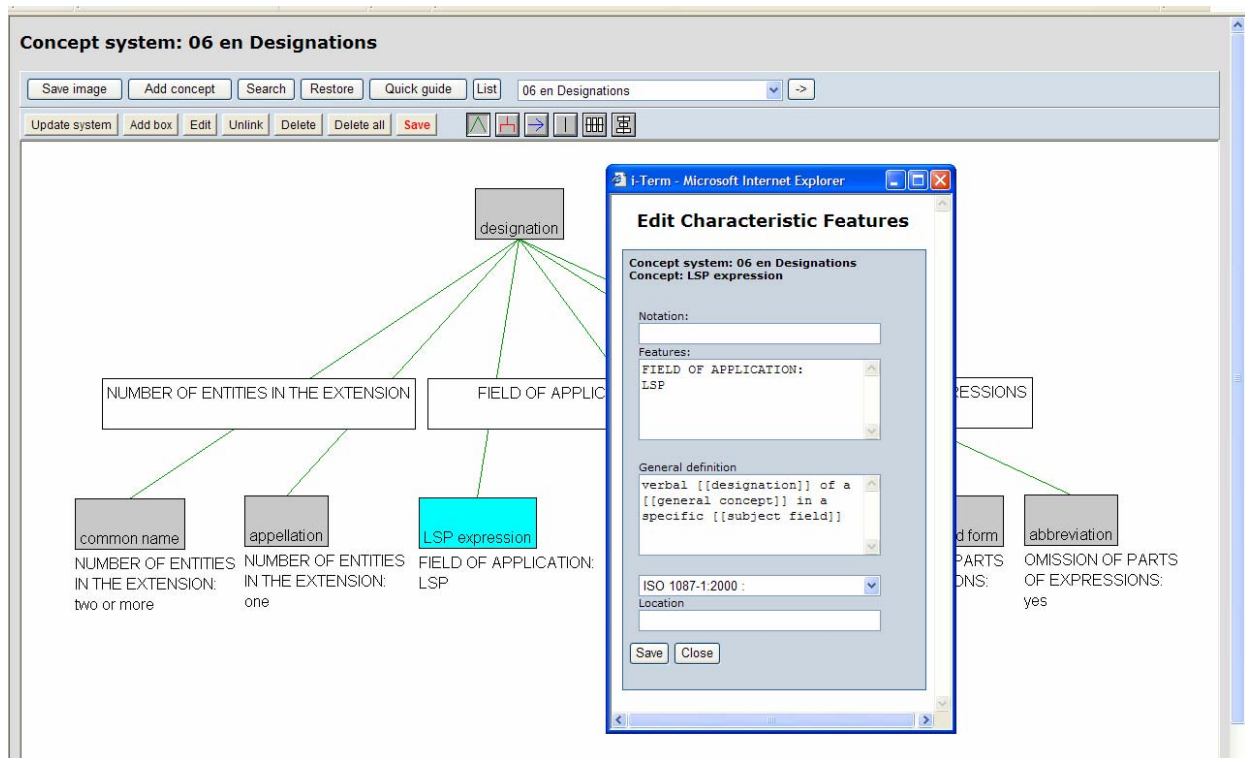


Figure 11: Editing characteristics (feature specifications) and definitions in i-Model

*appellation* in the Swedish concept system, are only represented in the system for concept types in Danish.

The concepts *LSP expression* and *general language expression* in the Danish system differ with respect to field of application. *LSP expression* (Danish *fagudtryk*) is equivalent with the Swedish concept *term*. The Danish definition of *term* is a “linguistic sign used in LSP that is a combination of content (concept) and expression”. In a note it is mentioned that the expression *term* is used both for the linguistic sign (i.e. expression + content) and for the expression alone, but that in Danish it is recommended to use *term* only for the linguistic sign.

The final distinction between *abbreviation* and *non-abbreviated form* in the Swedish system in Figure 4 is identical to the distinction in the Danish system.

Figure 12 shows the Swedish concept system for concept types (created in i-Model using English translations of the terms). The Danish concept system for concept types in figure 9 differs from the Swedish

system, probably due to different traditions in the two countries. Below we will go more into detail on the differences in the two concept systems. The use of concept systems and feature specifications help the terminologists in clarifying the differences and in deciding whether a harmonization would be feasible.

In the Swedish system, *superordinate concept* and *subordinate concept* are defined as concepts in a generic relation (they differ with respect to their position in a generic concept system). However, in the Danish concept system these are not defined as concepts in a generic relation. They are defined with respect to their position in a concept system, be it in a generic or a partitive concept system. In fact the Danish version of the concept system corresponds to ISO 1087-1 (2000) with respect to the definitions of *superordinate concept* and *subordinate concept*.

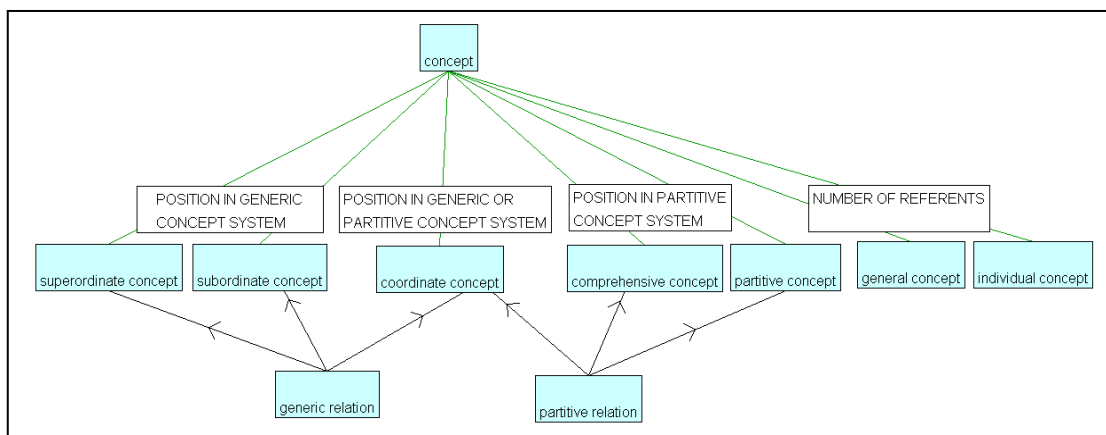


Figure 12: Swedish concept system for concept types

The Swedish definition of *superordinate concept* corresponds to the definition of *generic concept* in the Danish concept system and in ISO 1087-1(2000): concept in a generic relation having the narrower intension (“begrepp som står i generisk relation till ett annat begrepp och vars intension inkluderar i intensionen av det andra begreppet”).

In the Danish concept system you find the three concepts *concept in generic relation*, *concept in part-whole relation* and *concept in associative relation*. These concepts are differentiated with respect to relation type. By introducing these concepts, the four Danish concepts *generic concept*, *comprehensive concept*, *specific concept* and *partitive concept* all have two superordinate concepts. This means that they are defined with respect to both their position in a concept system and the relation type (each of them inherit characteristics from two superordinate concepts, and they are differentiated from each other by means of the combination of two characteristics). These four concepts correspond to the concepts with the same designations in ISO 1087-1 (2000).

One may argue that the designations of the three concepts added (*concept in generic relation*, *concept in part-whole relation* and *concept in associative relation*) are not used very often. But logically the concepts exist, and in our work in the CAOS project we use this kind of concepts when describing the principles and constraints that the CAOS system builds on. In the above mentioned definition of *generic concept* in ISO 1087-1(2000) the genus proximum is in fact: *concept in a generic relation*.

In ISO 1087-1 (2000) the concept *superordinate concept* has the following definition: concept which is either a generic concept or a comprehensive concept. This definition is not ideal since it is an extensional definition. This kind of definition may be avoided when building the definition on the feature specifications in the Danish concept system. The definitions of *generic concept* and *comprehensive concept* correspond to the Danish definitions that are based on the feature specifications.

In the first version of the Danish concept system the subdividing criteria for the concepts *superordinate concept*, *coordinate concept* and *subordinate concept* position in concept system was position in a hierarchical concept system. However this was abandoned, because a concept system, in which one concept may have several generic concepts, is formally not a hierarchy. However, the concepts are related by means of the generic relation, and therefore it would be too restrictive to define the generic relation as a hierarchical relation.

## 5. Special needs identified during the project

Both i-Term and i-Model have been and are for the moment being further developed to meet the special needs that have been identified in the NORDTERM project described here. For example it is very important in this kind of co-operation, where also several people in the local groups enter and edit data, to keep track of changes, and to make it possible for the terminologists

to enter editorial comments, that can not be seen by end users. Another very important facility is to allow systematic lists of concepts. This facility has already been developed, cf. figure 10. Also it is important to be able to sort the terms in the result overview and in export files according to the rules of the individual languages. Another example is the development of more advanced cross link facilities, allowing links between terms that are not in the base form.

## 6. Conclusion

In this paper, we have described how the use of a web-based terminology management system has made co-operation between the project partners much more dynamic than earlier, when new versions of Word-files with definitions etc. were circulated for comments, and all participants had to check whether their latest corrections had actually been included. There is no longer a need for special procedures for publication, and the NORDTERM partners may access the database directly instead of accessing PDF files or static HTML pages.

We have also shown how the methodology of adding characteristic features to the concepts may help the terminologist to build logically consistent concept diagrams.

## 7. Acknowledgements

The NORDTERM project: Terminology of terminology in Nordic languages received resources from Nordplus Sprog (Nordplus Language: <http://www.ask.hi.is/page/nordplussprog>) for the period 2004-2005.

In 2006 Nordplus Sprog granted resources for the use and further development of a system for management and publishing of the results of the project, including conversion and import of the vocabulary data.

## 8. References

- Madsen, Bodil Nistrup, 2005. “Implementation Of Research Results In The Development Of An Internet Terminology And Knowledge Management System”. In: Nina Pilke, Birthe Toft (eds.) Proceedings from *Terminology in Advanced Management Applications. 7th International Conference on Multilingual Knowledge and Technology Transfer*. Terminology Science and Research, Vol. 16 (2005).
- Madsen, Bodil Nistrup, Hanne Erdman Thomsen & Carl Vikner. 2005. “Multidimensionality in terminological concept modelling”. In: Bodil Nistrup Madsen, Hanne Erdman Thomsen (eds.): *Terminology and Content Development*, TKE 2005, 7th International Conference on Terminology and Knowledge Engineering, Copenhagen: 161-173.
- NORDTERM 2, Terminologins terminologi. NORDTERM: 1989.
- ISO 1087-1:2000 Terminology work -- Vocabulary -- Part 1: Theory and application. 2000.

# Engaging Customers in Terminology Creation

## For Windows Vista

Barbara Inge Karsch

Microsoft Deutschland GmbH  
Konrad-Zuse-Straße 1  
85716 Unterschleißheim  
Germany  
Bkarsch@microsoft.com

### Abstract

Terminology management for software products is an intricate affair: Large numbers of concepts need to be turned into terms that guarantee usability of the product and meet expectations of a variety of users. While responses by German users regarding the linguistic presentation abound, participation of users in the creation process is not very common. This paper describes the involvement of Microsoft® partners in term creation for the German version of the next operating system. This product-specific terminology process has several different stages: the creative phase which involved customers; the standardization phase performed by the marketing, translation and terminology teams; and the dissemination of terms to terminologists, localizers and translators. Although the engagement with users for term creation was not concluded by the submission deadline, several concepts and terms are discussed in detail in this paper.

## 1. Introduction

In 2005, the first issue of eDITion, a German terminology magazine (Deutscher Terminologie-Tag e.V. and Deutsches Institut für Terminologie e.V. 2005) was devoted to the “language of the customer.” The magazine focuses on cost savings and translation quality improvement through terminology management. But it also covers the strategic aspect of speaking the customers’ language (Childress 2005). Childress states that “well informed customers, who are interested in the company and the company’s products and services” should be involved in the terminology strategy. This paper will give an example of a project where customers were directly involved in creating and evaluating terminology.

Where they do exist, terminologists have always had an eye on the clients’ vernacular: They read trade journals, consult with subject matter experts, and participate in technical training. Through customer surveys, consultant feedback or term adoption rates<sup>1</sup>, they find out whether their choices met expectations. The more established a field the more emphasis is placed on what could be considered side shows, such as the linguistic presentation of a product. The software industry has unquestionably advanced to that level of customer focus: user experience has become one of the center points in software development. And more and more customers are involved before, during, and after the development cycle.

Apart from the project management challenges, it is not a small feat to create linguistically appropriate versions. The new Microsoft operating system, called Windows Vista™, will be localized into almost 100 languages. For languages of limited diffusion, the translation scope is smaller, i.e. smaller language packs of the operating system will be offered, however in these languages there may not even exist IT terminology to use. In the framework of the Local Language Program (LLP),

Microsoft has coordinated the creation of glossaries for languages such as Greenlandic, Māori or Nepali, through the community. Any volunteer from the respective linguistic group can join the program, make suggestions, add comments, or make the case for their translation. A moderator reviews and selects the most appropriate translation (Microsoft Corporation 2006).

For languages, such as Japanese and German, the extent of localization is significantly larger. While a substantial percentage of the terminology is established, customer expectation is much higher. Terminological challenges are new concepts, names or technical terms without clear equivalent in the target language.

For Windows Vista, the languages covered by MILS internally (Brazilian-Portuguese, French, Italian, Japanese, Korean, Traditional Chinese, Russian, Simplified Chinese, Spanish) are involving their communities. After a short theoretical discussion, this paper describes the process of community engagement in terminology creation for the German version of Windows Vista. In particular, it will describe the community involved as well as the process flow. The results, which are not 100% final, will be discussed and evaluated as far as possible and a conclusion will be drawn.

## 2. Term Formation

Term formation is “the process of naming the concepts required by a particular special language community for the development of cognitive processes and communication, [and is] a conscious human activity based on the awareness of pre-existing models.” (Wright 2003)

To excel in this process, a terminologist must know their language community. The customer base of Windows Vista is extremely heterogeneous: There are expert users with computer degrees who may install systems for thousands of people, and there are computer novices (see also (Childress 2005)). The objective that communication be established through well-chosen terms means usability in the context of software. Localization has the additional challenge that real estate on the screen

<sup>1</sup> Adoption can be gauged by the number of hits on the internet, for example.

is limited. The awareness of existing terms and concepts requires that terminologists have experience in the subject field and master their research methods.

Concepts can only be named, if they have been identified. Since this is a discussion of secondary term formation, i.e. of creation of foreign equivalents, existing English terms were extracted manually and automatically.

The input of customers and experts is one important research method. It is of invaluable help, if terminologists can form or have access to a community of practice, i.e. “groups of people with similar interests doing things together to achieve some end” (Addleson 2000).

In software technology, concepts require very different levels of expertise and terms belong to different registers (see also (Sager 1990), page 80). Terminologists must use their communities wisely. For instance, it would make little sense to ask an average computer user how to translate the technical term “persistence” into German, but it is very useful to ask whether he prefers *downloaden* or *herunterladen*.

### 3. The MVP Program

The community of practice chosen for this project is the Microsoft MVPs. “Microsoft’s Most Valuable Professionals (MVPs) are recognized, credible, and accessible individuals with expertise in one or more Microsoft products and technologies who actively participate in online and offline communities to share their knowledge and expertise with other Microsoft customers.” (Microsoft Corporation 2006). The MVP program is a worldwide program and currently has 954 members in EMEA<sup>2</sup>, 144 of them are part of the German MVP program. Upon suggestion by peers or Microsoft, membership is awarded for a period of 12 months. The membership and engagement within the MVP Program are of voluntary nature. MVPs have the opportunity to receive information and disseminate it in their communities; moreover, they are offered feedback channels to Microsoft on what their community would like to see in new releases. MVPs may thus support information flow from Microsoft to the end user and back: through conferences, surveys, discussion forums and informal communication.

MVPs are an ideal group for this particular project, as they all are awarded for their technical expertise, e.g. Windows, and encompass a wide range of professions; among them are developers, consultants or enthusiastic power users. All of them are in contact with other users, who range from home office users to CIOs. MVPs are familiar with the level of software literacy in their community and speak their language.

Of course, MVPs are also creative humans, and they may or may not apply principles of good term creation. If they do, they will most likely do so subconsciously. Creativity, an important aspect of term formation, is well applied early on in the process. The remaining steps will have more of a standardizing effect on any terms that might not be suitable (see also (Karsch 2006)).

### 4. The Process

Localization of an operating system is not a small project. The software entails thousands of terms, many of

them established, some of them new. In the last few years, Microsoft has made significant strides toward integrating terminology management in the product life cycle (compare (Irmeler 1999), page 519ff). The internal terminology group, Microsoft International Language Solutions (MILS), in cooperation with in-house localizers extracted about 800 new terms from the software files. These terms were defined and stored in the terminology database.

MILS terminologists are in charge of coordinating the localization of terminology before the actual start of translation. During this process, they fall back on their pool of experts. For German, we decided to involve the MVPs for a subset of 39 terms. We chose:

- Feature names
- Technical terms
- Names of games

The terms were provided together with definitions and screenshots, but no German suggestion, and posted on a Windows® Sharepoint Services share. This share is accessible to users with an account. MVPs were given an account and invited to suggest translations. They were also asked to sign non-disclosure agreements (NDAs), since terms and concepts are of confidential nature until the first beta versions are released or marketing campaigns start.

After the nine participants concluded their brainstorming, the suggestions were discussed by the following stakeholders: Marketing staff who focused on high-visibility names for their commercial effect; subsidiary program managers and localizers evaluated the terms with regard to technical accuracy, translation and/or technical implications; and the terminologists ascertained that all aspects of good term formation were covered. The finalized terms were entered into the terminology database and applied by in-house localizers and external translators.

### 5. The Results

At this point, 34 out of a submitted 39 concepts have been named. Of these 34 terms, 19 were taken verbatim from and 10 were based on MVP suggestions. That means that 85% originate from customer ideas. It would be beyond the scope of this paper to deliberate all 34. Therefore, the following sample terms were elected for this discussion: glassiness, gadget, health, Purple Shop, Windows Easy Transfer. The next sections describe the concepts, list the suggestions of the MVPs, and then explain the decisions by marketing, localization and terminology experts.

#### 5.1. Glassiness

Definition: “The appearance of the Start menu and the taskbar after the glass visual style has been applied to customize the color, intensity, and opacity of window borders.” In addition, the MVPs were provided with a screenshot that illustrated the concept.

Suggestions from MVPs:

- Transparenz (noun, mass)
- gläsern (adjective)
- Durchsichtigkeit (noun, mass)
- Gläsernheit (noun, mass)
- Milchglass-Effekt [sic] (noun, mass)
- Glas-Effekt (noun, mass)
- Glaseffekt (noun, mass)

<sup>2</sup> Europe, Middle East and Africa

Glassiness represents a new concept in Windows, which is intended to render the interface interesting. Localizers, marketing staff and terminologists unanimously agreed on *Glasseffekt* as it covers the characteristics “appearance” and “glass,” can be applied similarly in German contexts as the English equivalent and is linguistically correct. Furthermore, it has a more novel sound to it than, for instance, *Transparenz*.

## 5.2. Gadget

Definition: “A mini-application in Windows Vista that is designed to provide information, useful lookup, or enhance an application or service to a user’s computer. Examples of gadgets: a weather gadget running on your desktop or on your homepage and displaying weather in the cities of user’s choice; an RSS Gadget that pulls in user’s favorite feeds; or an extension of a business application providing just-in-time status on the pulse of user’s business.” The MVPs were provided a screenshot for this concept as well.

Suggestions from MVPs:

- Häppchen (noun, singular)
- Sitebar Komponente [sic] (noun, singular)
- Wichtel (noun, singular)
- Gadget (noun, singular, Anglicism)
- Teile (noun, plural)
- Stücke (noun, plural)
- Helferlein (noun, singular, diminutive)
- Heintzelmännchen (noun, singular, diminutive)
- Anzeigemodul (noun, singular)
- Tool (noun, singular, Anglicism)
- Minianwendung (noun, singular)

One MVP also remarked that the Anglicism *Gadget* was known to technically versed users in Germany. He recommended not using it. Furthermore, he stated that an earlier incarnation of the concept was called “Sidebar Tile” and suggested playing with the German equivalent of “tile.”

The contributions for this concept were very creative. The decision was made in favor of *Minianwendung*. This German term is more sober than the English “gadget” which is a very widely applied term and implies play, toy, etc. Several of the suggestions (e.g. *Heintzelmännchen*, *Helferlein*, *Wichtel*) carried this notion. *Tool*, *Anzeigemodul*, *Stücke*, and *Teile* were too generic to express the concept and could have been confused with existing concepts. *Minianwendung* is safe, as it is the translation of the superordinate; the distinguishing characteristics are lost, but can be derived from context in most cases. Furthermore, users can be expected to associate the appropriate functionality with this term.

## 5.3. Health

Definition: “A specification of the required conditions for full access. Health policies are configured in IAS. A network might have more than one health policy. For example, DHCP Quarantine and VPN Quarantine might use different health policies.” We sought feedback on this known technical term, because no one German term had emerged as the leading translation.

Suggestions from MVPs:

- Umgebung (noun, singular)
- Umwelt (noun, singular)

- Gesundheitszustand (noun, singular)
- Wartungszustand (noun, singular)
- Zuverlässigkeit (noun, mass)
- Sicherheitszustand (noun, singular)
- Wartungsstand (noun, singular)

This decision is not easy and is still not 100% final. The current front runner is the literal translation *Gesundheit*. In essence, “health” is part of a large conceptual system with many compound nouns (e.g. health policy, Statement of Health, Statement of Health Response, System Health Agent, System Health Validator) that any of the more descriptive German compounds would have rendered consequent term formation difficult. While users may initially be surprised to read about *Gesundheit* in the context of computers, the expectation is that they will understand the concept quickly and get used to the term, just as they did with *Virus*.

## 5.4. Purble Shop

Definition: “A game that is part of the Purble Place game that allows the player to build a face.” There are several games that belong to this conceptual system.

Suggestion from MVPs:

- Schminkecke (noun, singular)
- Schminksalon (noun, singular)
- Purble Shop (noun, singular, Anglicism)
- Wie seh’ ich wohl aus? (phrase)
- Purble (Schmink)Salon – Dein eigener Schminkkasten! (phrase)
- Gesichterraten (noun, mass)

Here, too, submissions were very creative. The final decision was largely based on the fact that Microsoft Deutschland consistently receives feedback from general users to avoid Anglicism. Since the games are for a general as well as younger audience, a German term was chosen. After the MVP phase was concluded, more information became available and the definition was revised to include not only faces, but whole characters that can be built with this game. We therefore decided for *Figurenraten*, which does not quite accurately reflect the concept. *Raten* means to guess, where the game actually lets the user build something.

## 5.5. Windows Easy Transfer

Definition: “A Windows Vista tool that assists users in moving their data (documents and settings) to the new environment.”

Suggestions from MVPs:

- Umzugsassistent (noun, singular)
- Persönliche Einstellungen und Dateien übertragen (phrase)
- Dateien und Einstellungen auf einen anderen Rechner übertragen (phrase)
- Windows Migrationsassistent (noun, singular)
- Windows Transfer (noun, mass)
- Windows-Übertragung leicht gemacht (phrase)
- Assistent zum Windows-Umzug (phrase)
- Übertragen von Einstellungen und Dateien auf einen anderen Computer (phrase)
- Windows Einstellungs-Übertragung (noun, singular)

This concept already existed in Windows XP and is called “Files and Settings Transfer Wizard” (*Assistent zum Übertragen von Dateien und Einstellungen*). For this rather technical feature surprisingly, only one suggestion was based on the English term. All others were more or less resourceful renditions of the English meaning. In this case, the marketing team drove the decision for the English term *Windows Easy Transfer*. Since *Transfer* is a German word with a similar meaning and the adjective *easy* has become part of German slang, this Anglicism is probably not hard to understand by average computer users who install the operating system themselves.

## 6. General Evaluation and Conclusion

We expected this process to result in creative, accurate, and user-friendly terms. When terms are produced “artificially” by a terminologist who covers dozens of terms per day from various products and subject matters, without doubt creativity can get lost. The MVPs were uninhibited, as it were, even in creating new technical terms (see the colorful suggestions for “gadget”).

For this purpose, we declare a term as accurate, if it reflects the meaning of the concept, is grammatically and orthographically correct, and linguistically fits into the system of the target language (see “health”). This presupposes that concepts are described correctly by their definitions. Since functionality changes until late in the software creation process, correct definitions are not a given.

A focus on user-friendliness was built into this process, more than is the case in typical terminology creation processes. Furthermore, final terms were deliberately matched to the target audience. Whether the suggested terms will be accepted by the users, remains to be seen. Success in this area could be evaluated by surveys after release to market.

This process could be compared to funneling material through a standardization and quality assurance process: many terms devised by experts with lots of creativity are poured into the funnel, examined from various angles, discarded or selected, so that one standardized term per concept emerges in the end.

Technological advances in the last 15 years have drastically altered the accessibility of terminological resources. In the past, translators were permanent residents of libraries and had large circles of friends in various and sundry professions; today, they have a computer. Restricted access through SharePoint software allows partners to contribute quickly and efficiently, even if they are in Thailand (as was the case for one MVP), the terminology team in Germany, and the localizers in the United States. Localizers can retrieve prepared terminology in an online centralized database that is maintained around the clock.

This is not the first time that MVPs graciously devoted time and brain power to Microsoft terminology. In early 2005, Visual Studio terms were discussed by MVPs. The administrative effort to arrive at 39 terms is fairly high. Considering the volume of terms that two terminologists must cover for approximately 450 product teams in just one language, this process is fairly luxurious. But the expectation is that adoption by the German public will be

high. This would result in less rework and therefore in time and cost savings.

## 7. References

- Addleson, M. (2000). Organizing to Know and to Learn: Reflections on Organization and Knowledge Management. In *Knowledge Management for Information Professionals*, T.K. Srikantaiah and M.E.D. Koenig (Eds). Medford, NJ: Information Today, Inc. p. 137-160.
- Childress, M. (2005). Die Kundensprache In *eDITion*. 1. p. 4-7.
- Deutscher Terminologie-Tag e.V and Deutsches Institut für Terminologie e.V. (2005). *Die Sprache des Kunden*. eDITion - Terminologiemagazin. J. Zeumer (Ed). Cologne: SDK Systemdruck Köln.
- Irmeler, U. (1999). Terminology Management in Software Localization: Standardization and Its Limits. In *Terminology and Knowledge Engineering: Proceedings TKE '99*, P. Sandrini (Ed). Vienna: TermNet. p. 518-527.
- Karsch, B.I. (2006). Terminology workflow in the localization process. In *Perspectives on Localization* (in production), K.J. Dunne (Ed). Amsterdam: John Benjamins Publishing Company.
- Microsoft Corporation (2006). *Microsoft Local Language Program*. Microsoft. Last accessed: Feb 17, 2006. Website: <http://www.microsoft.com/industry/government/locallanguage.msp>.
- Microsoft Corporation (2006). *Most Valuable Professional*. Last accessed: Feb 17, 2006. Website: <http://mvp.support.microsoft.com/>.

# Some notes about the evaluation of terms and term extraction systems

Jorge Vivaldi\*, Horacio Rodríguez#

\*Institute for Applied Linguistics, Universitat Pompeu Fabra  
La Rambla 30-32, 08002 Barcelona, Spain  
jorge.vivaldi@upf.edu

#Software Department, Universitat Politècnica de Catalunya  
c/ Jordi Girona 31, 08034 Barcelona, Spain  
horacio@lsi.upc.es

## Abstract

Term extraction may be defined as a text mining activity whose main purpose is to obtain all the terms included in a text of a given domain. Since the eighties there has been a growing interest on it, mainly due to the rapid scientific advances as well as the evolution of the communication systems. During this time, a number of techniques and strategies have been proposed for satisfying this requirement. At present it seems that term extraction has reached a maturity stage. Nevertheless, many of the systems proposed failed to qualitatively present their results, almost every system evaluates its abilities in an ad-hoc manner (if any, many times). Often, the authors do not explain its evaluation methodology, therefore comparisons between different implementations are difficult to draw.

## 1 Introduction

Term extractors (TE) have been mostly developed to solve some specific necessities; (see some of them in Cabré et al., 2001); that is, they have been developed for a specific domain and language and to fulfill some particular need such as: glossary compilation, translation or some NLP purpose (as information retrieval, ontology/conceptual maps generation, etc.) among others. For this reason, perhaps, little effort has been done for reaching some kind of consensus in a standard evaluation procedure. But another problem arises in the process of evaluating a TE: who determines which are the terms in a given test text? This issue arises because two different actors with different profiles are involved: a terminologist, expert on deciding whether an expression is a real term or belongs to the general language, and a domain expert, who uses a specific expression to refer to a concept in the domain. This point has often not been taken into consideration.

Terms are usually defined as lexical units used to designate concepts in a thematically restricted domain. The detection of these units is a difficult and complex task and, as mentioned in Sager (1999), such complexity is mainly because “terms adopt all the word formation rules in a given language”. Also, as mentioned in the term definition itself, it is necessary to assure that a given lexical unit belongs to a specialized domain. Due to the difficulties to verify this condition, we usually refer to the result obtained by a TE as “term candidates” (TC) instead of just “terms”.

In this paper, we plan to discuss some issues about the task done for evaluating YATE, a hybrid TE. This evaluation schema could be easily applied to other TE. After this introduction, the organization of the paper is the following: Section 2 presents a brief introduction to term extraction evaluation methods. Section 3 describes several approaches to the problem in some well known term extraction systems. Section 4 briefly introduces the system evaluated and describes with some detail our proposal. Sections 5 and 6 describe the evaluation strategies followed to evaluate both terms and term extraction system. Finally, in Section 7, we present our conclusions.

## 2 Term extraction evaluation basics

Almost all TE systems have their origins either in Information Retrieval or Linguistics. The former has based its evaluation measures in the precision and recall measures while the latter is based in the noise and silence figures. Both perspectives give basically the same information in a different way. They may be calculated as shown in Figure 1.

The measures chosen to evaluate YATE, a TE described in section 4, were precision and recall as defined in the Information Retrieval area. Precision measures the degree of correctness of the TC that are proposed as terms while recall measures the degree of comprehensiveness of such TC. Usually both figures must be considered together to give consistency to the conclusions drawn. As we will show in section 5, recall is the hardest figure to calculate since it implies to know in advance which is the whole set of terms included in the document under evaluation. The latter point introduces the problem, already mentioned above, concerning the actual number of terms included in the text under evaluation. As we will show in section 5, in practice it is difficult to define precisely what a term is. Additionally, it is worthwhile mentioning that all of the above mentioned behavior measures assume that it is possible to perform a binary decision.

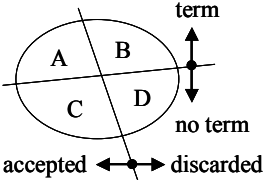
$$\begin{aligned} \text{precision} &= \frac{A}{A+C} \\ \text{recall} &= \frac{A}{A+B} \\ \text{noise} &= \frac{C}{A+C} \\ \text{silence} &= \frac{B}{A+B} \end{aligned}$$


Figure 1. Typical behaviour measures

We understand that the output of a TE should not be merely a list of term candidates chosen by some criteria but that such a list should be sorted according to their termhood. This figure has been defined in Kageura et al. (1996) as “the degree that a linguistic unit is related to domain-specific concepts”; therefore, it is appropriate for this task. Only in this way a user may decide, according to



his specific needs and resources, which percentages of the list should be accepted, revised and refused.

As more terms are extracted from the list of candidates recall increases while precision decreases. Several methods have been used in information retrieval (or related disciplines) to get a unique measure (F1 measure, cross points between recall and precision curves, average of precision for several recall points, etc.) Some of these methods seem to be appropriate for our purposes because the threshold for accepting the terms from the candidate list depends completely on the intended application. For this reason, to evaluate the result we decided to divide the sorted list of term candidates into a number of sections. Precision and recall computation is performed accumulatively for each section. Figure 2 shows the typical behavior of both an ideal and an actual TE. In the former case, the TE produces a list where all the terms are ranked in the top of the list; therefore, precision is kept at 100% for all the recall values. In the latter case, the TE fails to produce a sorted list of TC because non terms are inserted between terms; hence, precision drops as recall increases. In any case, a negative slope is the indicator that the TE is performing some kind of classification of the TC.

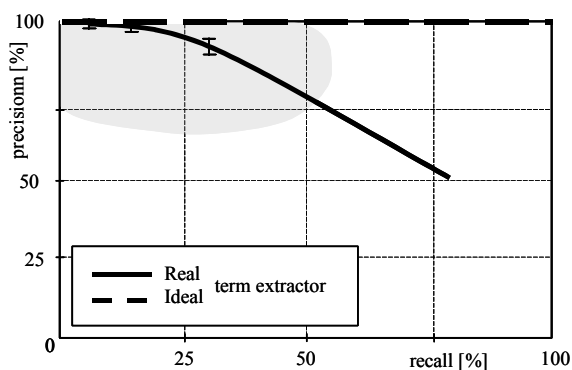
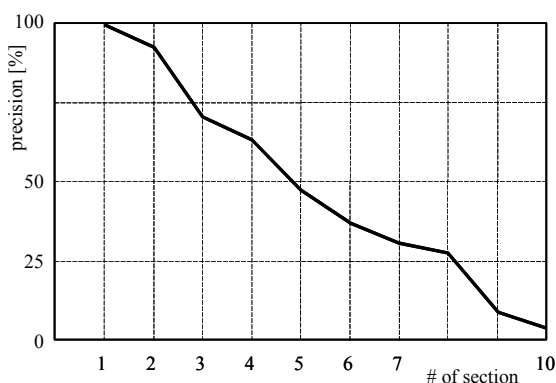


Figure 2. Typical behaviour using precision and recall

Due to the intrinsic difficulty in obtaining the whole set of terms in a document, some developers just calculate the precision. In this way, it is easier to ask a specialist about the termhood of a given TC than ask for all the terms in a given text. The cost of this simplification is losing information about the recall. In this case, the list of TC is divided in a number of sections and the accumulated



precision for all sections is given (see Figure 3).

Figure 3. Evaluation of a TE using only precision

In order to be able to appropriately assess a given TE an overall evaluation using a standard behavior measure is

obviously needed. Besides, depending of the internal structure of the TE under analysis, its internal modules should be also evaluated. Only in this way it is possible to find the weak points of the system and improve the final results.

Some of the evaluation problems found in TE also occur in other NLP areas and have led to evaluation proposals as ROUGE in Automatic Summarization (see Lin, 2004), or BLEU in Machine Translation (see Papineni et al, 2002). Recently there have been some attempts to integrate different measures coming from several sources or evaluators in order to increase the robustness of the evaluators. ORANGE (Lin, Och, 2004) and QARLA (Amigó et al, 2005) are two interesting approaches.

### 3 Related approaches

Perhaps the first serious evaluation criteria have been shown in L'Homme et al. (1996) where a quite simple method was proposed: "to compare a list of terms produced by a human to the one produced by the system". It is very interesting to note that the authors state that "humans may omit certain terms"; forewarning about the term evaluation problems and its peculiarities. The authors only mentioned noise and silence as the figures to evaluate the TE, revealing its linguistic origins.

LEXTER (Bourigault, 1994) is a well known linguistic TE system that has been designed taking into consideration a deep manual term validation process. They only report the validation process through a silence of about 5%.

TRUCKS is a hybrid TE system described in Maynard (1999) and is one of the TE that has been more exhaustively evaluated. Maynard tried to use UMLS<sup>1</sup> as a list of predefined terms but failed, mainly, as she pointed out, due to the incompleteness of such resource. Finally, the author carefully evaluated her system against a manually build list. This system applies a number of measures consecutively. She assessed the system in a global way and its main components using precision and recall as evaluation measures. The author recognized the difficulty in determining the terms included in a text; two experts collaborated in this case but no comparison was made among them.

FASTR (Jacquemin, 2001) is a true linguistic TE system although rigorously speaking it is a tool for detecting variants that may be used as a TE. Jacquemin started from a large terminological list of several domains provided by the INIST documentation center. His approach focuses mainly on how to enlarge this list through corpora observation. Basically, the system starts with a small group of seed terms that allow to detect new terms that are added to the seeds. These new terms are not evaluated, so a wrong term may potentially originate new wrong terms. The only measure taken into consideration is precision.

Another system that has been carefully evaluated is TermStar, described in Drouin (2002). The methodology here foresees two stages: the first one is automatic (against a term bank) and the last one manual (by a number terminologists). In this case, the precision and recall figures are calculated by grouping the TC by frequency giving a global figure for each set. This calculation procedure allows to see the influence of the TC frequency in the final result. Additionally, all the recall calculations

<sup>1</sup> <http://umlsinfo.nlm.nih.gov>

have been made taking into account the manual evaluation with the aid of the TE itself (it means that it was done not by direct text reading).

There are many others TE where the information regarding evaluation is limited to mentioning a “global” figure of precision (that is, there is no mention about the corresponding recall). Additionally, in some other systems the figures reported are obtained with a text of a few k-words and sometimes no indication of the size is provided. From a different perspective it is interesting to mention the work done in the framework of the ARC A3 and (the undergoing) CESART projects (see El Haidi et al. 2004 for details). These projects deal with the production of French corpora for extraction tasks and elaboration of protocols to evaluate several terminological resource tools. The evaluation relies on human experts as this task can not be reproduced by automatic mechanisms. This project applies the black-box evaluation model and the precision and recall metrics.

#### 4 A term extractor

Since the 80s, multiple efforts have concentrated on term extraction. Many techniques have been developed but none of them in isolation has proved to be fully successful. Recently, the combination of different knowledge sources has proved to be successful in some NLP areas such as tagging, parsing and text classification among others.

The evaluation task described in this paper has been done on YATE (see Vivaldi 2001 for details), a term extraction tool whose main characteristics are: a) it uses a combination of several term extraction techniques and b) it uses EWN<sup>2</sup>, a general purpose lexico-semantic ontology as a primary resource. Taking into consideration that all the term extraction techniques used were heterogeneous we decided to apply a combination technique to this TE. Some of the aspects regarding this technique have been described in Vivaldi et al. (2001a, 2001b and 2002) where different ways of combination are presented.

YATE was designed to obtain all the terms (from the following set of syntactically filtered candidates: <noun>, <noun-adjective> and <noun-preposition-noun>) found in specialised texts within the medicine domain. As mentioned above, YATE is a hybrid TE system that combines the results obtained by a set of term analyzers described briefly as follows:

- domain coefficient (MC): uses the EWN ontology to sort the TC.
- context (CFp): evaluates each candidate using other candidates present in its sentence context.
- classic forms: it tries to decompose the lexical units in there formants, taking into account the form characteristics of many terms in the domain.
- collocational method: evaluates multiword candidates according to its mutual information.

The results obtained by this set of heterogeneous methods are combined using two different methods: voting and boosting. In the former each single term analyzer reports a term/no term status while the latter makes use of a well-known method originated in the machine learning area.

To evaluate this TE, we have used two Spanish texts from the IULA Technical Corpus (see Badia et al. 1998 for a

description). The first document (a) with a size of about 100K words includes a collection of medical reports while the second one (b) with an extension of 10K words was taken from a university textbook. Both documents may be considered, according to the classification established in Pearson (1998), as “highly specialized” and have been evaluated by some specialists who found 1446 and 699 terms for each<sup>3</sup>.

The first document has been used mainly for training while the second one has been used for testing. Due to the characteristics of the boosting combination method it has been applied only to the longest document using 10-fold cross validation<sup>4</sup>.

Although YATE has been developed on the medical domain, it can be tuned to other domains. Adaptation to Genomic, Law and Economics are undergoing projects.

#### 5 Terms evaluation

One of the reasons why term extraction systems are so important is because manual extraction is not only a time-consuming and labor-intensive activity but also because it tends to be inconsistent and therefore not reliable. An alternative may be to consider as terms only those string included in some pre-established list of known terms (like UMLS in Medicine). Unfortunately, such resources are far from exhaustive and they are almost not available for languages other than English.

Thus, the only solution to the problem of evaluating a TE is to manually find all the terms present in a given test text. As far as this paper concerns, this task has been made by domain experts<sup>5</sup>. A difficulty immediately arises: there are discrepancies about what must be considered as a term.

This difficulty has two closely related viewpoints: from one side it is difficult to decide if a given lexical unit is a term or not. On the other hand there may be disagreement between terminologists and domain experts but also among the latter and, in general, between any kind of terminology users (for a discussion, see Estopà, 1999).

To confirm the lack of consensus among specialists we will show the results found in evaluating the terms chosen in the document (b) of evaluation (10K words of a university text book). In Table 1 we show the global result for each evaluator for those terms that follows the noun pattern.

From all the terms chosen by three specialists, there was full agreement only for 37% of the terms, there was agreement between two specialists for 26% of the terms

<sup>3</sup> These quantities may be classified according its linguistic pattern as follows:

Document	pattern			total
	noun	noun-adjective	noun-prep-noun	
(a)	696	664	86	1446
(b)	326	314	59	699

<sup>4</sup> In the 10 fold cross-validation the dataset is randomly divided into 10 sets with approximately equal size and class distributions. For each “fold”, the classifier is trained using all but one of the 10 groups and then tested on the unseen group. This procedure is repeated for each of the 10 groups.

<sup>5</sup> A domain expert in this context is defined as a person who has expertise in the domain (Medicine) and who is possibly involved in research activities in such domain. Such specialist evaluate the terms by reviewing the full text of each document.

<sup>2</sup> <http://www.illc.uva.nl/EuroWordNet/>

and finally 37 % of the terms was chosen just for one specialist. In addition, some lexical units that were running as terms (like ‘epidemic’, ‘groin’ or ‘varix’) were not considered by any of the specialists.

evaluation	evaluator 1 (E <sub>1</sub> )	evaluator 2 (E <sub>2</sub> )	evaluator 3 (E <sub>3</sub> )
term	211	269	199
no term	115	57	127
total	326	326	326

Table 1. Result of the evaluation by each specialist

Due to the difficulties to reach an agreement about the actual number of terms, we decided to evaluate our TE taking as terms those linguistic units chosen by at least one evaluator.

The disagreement between the results obtained by different evaluators is common to other NLP activities that require manual validation like word sense disambiguation, discourse analysis or POS tagging. We compute the above mentioned agreement as usual, i.e. dividing the number of terms proposed by all the evaluators jointly by the total number of terms proposed by each of the evaluators (that is intersection/union). An issue with this score is that it does not take into account the agreement by chance. In Carletta (1996) and Ng (1999) it was suggested to use the Kappa coefficient as a reliability measure to take into account both true and chance coincidences. This measure is formally defined as follows:

$$\kappa = \frac{P_a - P_e}{1 - P_e} \text{ where: } P_a = \frac{A}{N} \quad P_e = \sum_{j=1}^M \left( \frac{C_j}{N} \right)^2$$

- and
- P<sub>a</sub>: standard probability of agreement
  - A: number of units where both evaluators coincide
  - N: total number of units
  - C<sub>j</sub>: number of term candidates that have been evaluated as j by some evaluator.
  - M: evaluation alternatives (1: term or 2: not term)
  - P<sub>e</sub> probability of agreement by chance

In using this figure to compare the results obtained by all three specialists for nominal term candidates we obtain the results shown in Table 2.

evaluators	Coincidence	P <sub>a</sub>	C <sub>cat</sub>	C <sub>nocat</sub>	P <sub>e</sub>	K
E <sub>1</sub> - E <sub>2</sub>	195	0,59	480	172	0,61	-0,05
E <sub>1</sub> - E <sub>3</sub>	223	0,68	410	242	0,53	0,31
E <sub>2</sub> - E <sub>3</sub>	194	0,55	468	184	0,59	-0,12

Table 2. Agreement between specialists evaluating nominal candidates<sup>6</sup>

Table 2 clearly shows some very low Kappa values; specially if we take into consideration that only Kappa values greater than 0.8 were considered in Carletta (1996) as optimal. This figure confirms the results obtained looking at the simple coincidence between specialists.

<sup>6</sup> C<sub>cat</sub> and C<sub>nocat</sub> stand for the number of TC that have been evaluated as ‘term’ and ‘not term’ respectively.

## 6 Evaluation of a term extractor

As mentioned in section 4, YATE is a TE that applies a combination technique to a term extraction task. We apply the evaluation methods described in section 2 to each extraction module and also each of their variants. In this paper we only show the best results obtained evaluating the patterns <noun> and <noun-adjective> with two combination methods (voting and boosting) and two single methods (domain coefficient and context factor). We do not apply boosting to document (b) because it is too small. We choose to work mainly on the above mentioned patterns because they cover most of the terms present in the documents.

Figure 4 and Figure 5 show the results obtained with document (a) and (b) respectively. Both figures consist of two parts: part a) shows the results for those TC that follow the noun pattern and part b) illustrates the results for the noun-adjective pattern.

A close examination of the results obtained for the noun pattern in document (a), shown in Figure 4a, allows to easily compare the results for different extraction strategies at any given recall level. For example, for a recall level of 30%, it is clear that the performance of both combination methods and a single method (Medical Coefficient) are similar and close to 100% precision. This precision level is about 50 percentage points (ΔP<sub>mv</sub> and AP<sub>mb</sub>) better than the other single method (CF, context factor). Improvement keeps the same at a recall level of 50% (ΔP<sub>mv</sub> and AP<sub>mb</sub>). Moreover, it can be seen that both combination methods perform similarly for recall levels till 30% but between 40% and 80% the boosting method performs better than voting. It is also interesting to observe that the MC method has a limited recall (about 45%) due to its peculiarities (it uses EuroWordNet, which has a limited coverage for Spanish and also it is a general purpose ontology, not an ontology specialized in the domain). The result obtained with the noun-adjective pattern for this document, shown in Figure 4b, clearly shows lower precision levels confirming that this pattern is harder to find than that for the nouns. Again, the results obtained for the combination methods are better than those obtained by single strategies and boosting is performing better than voting, this time for all recall levels. For a recall level of 30% the improvement of boosting and voting methods are high (ΔP<sub>mv</sub>=70% and AP<sub>mb</sub>=100% respectively). At 50% of recall level the improvement of combination methods keeps high (ΔP<sub>mv</sub>=53% and AP<sub>mb</sub>=87%).

The examination of the results obtained with document (b) allow to draw similar conclusions for both patterns. In this case, we only apply one combination method (voting) due to the fact that this document is much shorter and does not allow to apply the boosting combination method.

In section 5, we showed that the task of deciding whether a lexical unit is or not a term is a hard and subjective task. For this reason, we decided to check also the agreement of YATE against the results produced by the evaluators. The end purpose of this test was to confirm that the result obtained by YATE was closely related to the term list proposed by the domain experts. We grouped them in several ways: the union of all the evaluators, the union of the evaluators taken pair-wise, the intersection of all the evaluators and individually.

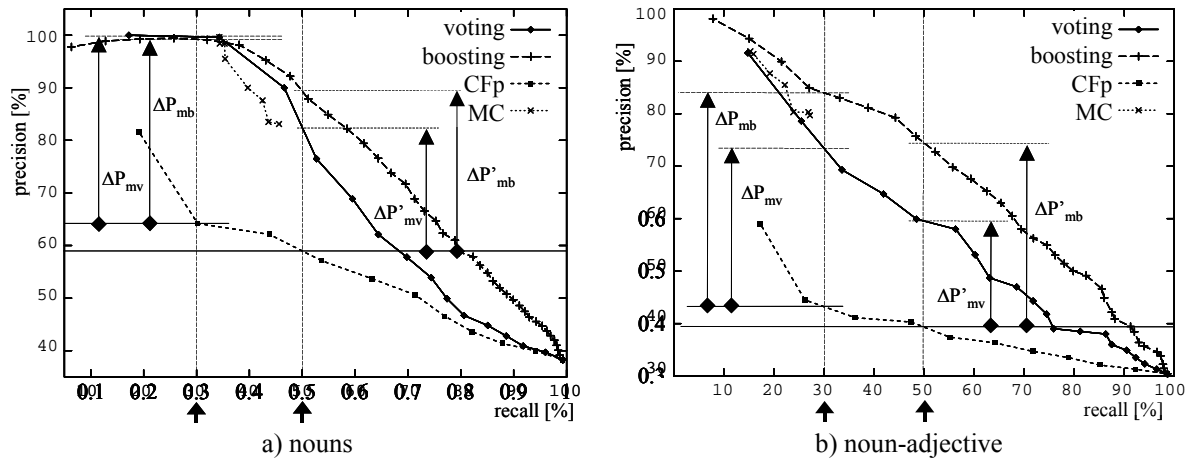


Figure 4. Results obtained for document (a)

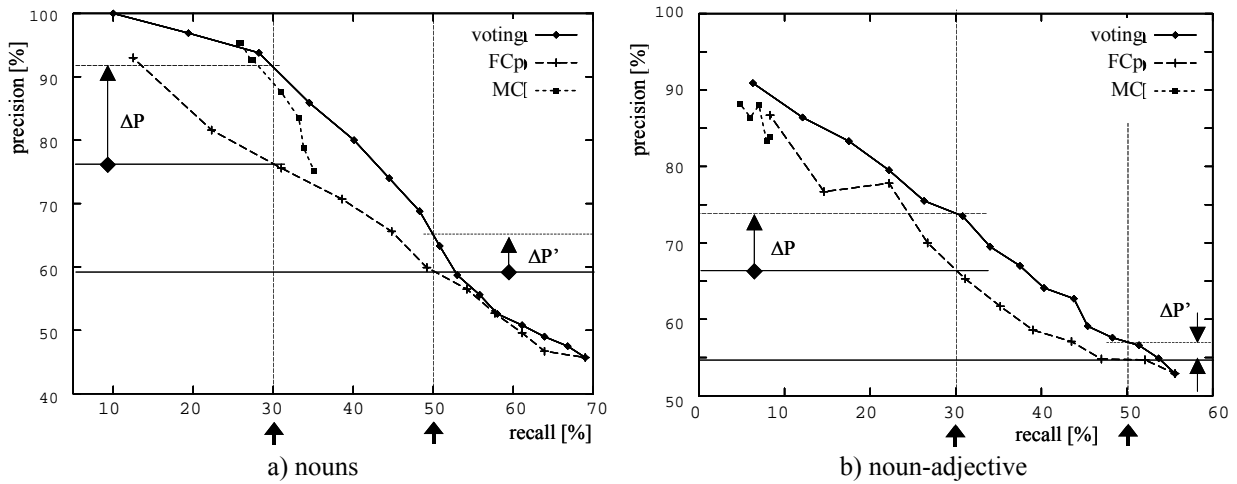


Figure 5. Results obtained for document (b)

For this task we used the Kappa coefficient as we did in section 5. We took the best result (voting) for document (a), divide the full list of TC in a number of sections and used the method sketched in Figure 6. Briefly, it means that we have a pointer and move it from left to right, keeping all the sections on the left as terms and all the right sections as non terms. For each of the (n-1) positions of the pointer and all the combinations of evaluators mentioned above we calculate the Kappa coefficient. The results obtained are shown in Figure 7.

Observing Figure 7 we notice that the maximum Kappa value is systematically obtained with the 20% of the sorted list of term candidates. Taking the evaluators individually, the Kappa value is between 0.6 and 0.65 rising to 0.75 when we take any combination of 2 evaluators. Taking all the terms considered by some of the evaluators Kappa reaches the maximum value of 0.82. We considered this result positively because taking the evaluators individually the Kappa is high (higher than comparing between evaluators), and it increases as the number of evaluators augments. The minimum value of Kappa is obtained choosing as terms only those units regarded as such by all the evaluators, which is a very restrictive condition.

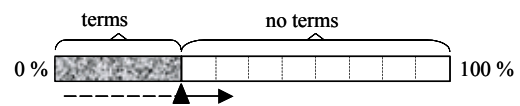


Figure 6. Mechanism to evaluate YATE vs. evaluators similarity

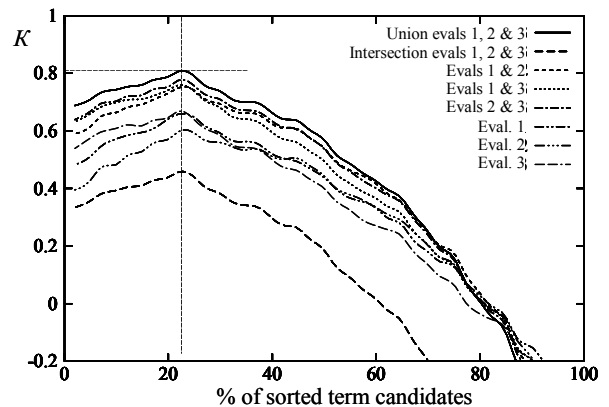


Figure 7. Comparison between YATE and the evaluators

## 7 Conclusions

This paper shows how a TE that uses hybrid techniques and has been developed for the medicine domain has been evaluated. For such a purpose, we have used the typical precision and recall measures taken from the information retrieval domain. We showed that observing these results, it is easy to compare different extraction methods in a clear and objective way.

We also show the difficulty to obtain the terms actually included in a given text. Such inconvenience is given by two different causes: from one side it is a manual task (therefore tedious and not error free) and from the other side, when it is performed by more than one specialist in the domain it may be difficult to obtain a consensus among all of them.

We consider that, in order to be able to appropriately evaluate a given TE and its internal modules, if possible, it is essential to make use of some standard performance measure. Taking into account that mistakes in the initial phases of processing (text segmentation, POS tagging, etc.) will influence directly the final results, it should be convenient to introduce a check point at the term candidates selection stage.

We believe, also, that the only way to do some progress in the term extraction area is to define, as in other areas of NLP, some kind of gold standard to check against. Such standard should include at least decisions about the corpus and its design criteria, metrics to be used and evaluation protocols for the terms included in the corpus. Due to the great variability of TE scenario and the low agreement between terminologists and domain experts on what candidates should be treated as terms, such gold standard should be highly parameterizable and should integrate (partial) evaluation pieces (and evaluators), following perhaps, ORANGE and QARLA ideas.

## 8 References

- Amigó E., Gonzalo J., Peñas A., and Verdejo F., (2005) QARLA: a Framework for the Evaluation of Automatic Summarization. In Proceedings of *the 43th Annual Meeting of the Association for Computational Linguistics* (ACL-2005).
- Bourigault, D. (1994), LEXTER, un Logiciel d'Extraction de TERminologie. Application à l'acquisition des connaissances à partir de textes. PhD thesis. École des Hautes Études en Sciences Sociales.
- Cabré, M. T., R. Estopà & J. Vivaldi, (2001). Automatic Term Detection: A Review Of Current Systems. In Bourigault, D., C. Jacquemin and MC. L'Homme (eds) *Recent Advances in Computational Terminology*. Chp 3. Amsterdam: John Benjamins
- Carletta, J. (1996) Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, Vol. 22 (2). Pages. 249-254.
- El Hadi W. M., Timimi I.& Dabbadie M. (2004). EVALDA-CESART Project: Terminological Resources Acquisition Tools Evaluation Campaign. In Proceedings of the *LREC2004 Fourth International Conference On Language Resources And Evaluation*. Lisbon. Pages 515-518.
- Lin Chin-Yew & F. J. Och (2004) ORANGE: a Method for Evaluating Automatic Evaluation Metrics for Machine Translation. In Proceedings of the *20th International Conference on Computational Linguistics (COLING 2004)*, Geneva, Switzerland, August 23- 27, 2004.
- Lin Chin-Yew (2004) ROUGE: a Package for Automatic Evaluation of Summaries. In Proceedings of the *Workshop on Text Summarization Branches Out (WAS 2004)*, Barcelona, Spain, July 25 - 26, 2004.
- Drouin P (2002). Acquisition automatique des termes: l'utilisation des pivots lexicaux spécialisés. PhD Thesis. Université de Montréal,
- Estopà R. (1999). Extracció de terminologia: elements per a la construcció d'un SEACUSE (Sistema d'extracció automàtica de candidats a unitats de significació especialitzada). PhD thesis. Universitat Pompeu Fabra.
- Kageura, K. & B. Umino (1996) Methods of automatic term recognition: A review. *Terminology*. John Benjamins Publishing Co., Vol. 3 (2). Pages. 259-289.
- L'Homme MC.; Benali L.; Bertrand C. & P. Laudique (1996). Definition of an evaluation grid for term-extraction software. *Terminology* 3:2, John Benjamins Publishing Co., Pages 291-312.
- Maynard, D. (1999) Term recognition using combined knowledge sources. PhD Thesis. Manchester Metropolitan University. Manchester.
- Ng H. T.; Lim C. Y. & Foo S. K. (1999). A Case Study on Inter-Annotator Agreement for Word Sense Disambiguation. Proceedings of *Siglex'99*. Pages. 9-13.
- Papineni, K.A., Roukos, S., Ward, T. & Zhu, W.J.(2002): Bleu: a method for automatic evaluation of machine translation. In Proceedings of the *40th Annual Meeting of the Association for Computational Linguistics*, 2002
- Pearson J. (1998) *Terms in context*. John Benjamin Publishing Co., Amsterdam.
- Sager, J. C., (1999). In search of a foundation: Towards a theory of term. *Terminology*, John Benjamins Publishing Co., Vol. 5 (1). Pages. :41-57
- Vivaldi, J., (2001). Extracció de candidats a término mediante combinación de estrategias heterogéneas. PhD thesis. Universitat Politècnica de Catalunya.
- Vivaldi, J. & Rodríguez, H, (2001a). Improving term extraction by combining different techniques. *Terminology*. John Benjamins Publishing Co 7:1., pages 31-47.
- Vivaldi, J. Márquez L. & H. Rodríguez. (2001b) Improving Term Extraction by System Combination using Boosting. In Proceedings of *the joint ECML-PKDD'01 Conference*. Freiburg, Germany.
- Vivaldi, J. & Rodríguez, H, (2002). Medical Term Extraction using the EWN ontology. In Proceedings of *Terminology and Knowledge Engineering (TKE2002)*.

# An application-oriented terminology evaluation: the case of back-of-the book indexes

Touria Aït El Mekki<sup>1</sup> and Adeline Nazarenko<sup>2</sup>

<sup>1</sup> LERIA

touria at info.univ-angers.fr

<sup>2</sup> LIPN UMR7030

Université Paris 13 & CNRS,

99 avneue J.B. Clément 93430 Villetaneuse, FRANCE

Adeline Nazarenko at lipn.univ-paris13.fr

## Abstract

This paper addresses the problem of computational terminology evaluation not per se but in a specific application context. This paper describes the evaluation procedure that has been used to assess the validity of our overall indexing approach and the quality of the IndDoc indexing tool. Even if user-oriented extended evaluation is irreplaceable, we argue that early evaluations are possible and they are useful for development guidance.

## 1. Introduction

Back-of-the-book indexes are precious information retrieval devices that offer an easy way to locate a given piece of information in a large document and to navigate through that document. Unfortunately, indexes are expensive to produce, because indexing remains mainly manual. Modern word processing or indexing tools provide a technical assistance but do not address the index content and information selection problem. The professional indexing tools (Sonar BookEnds, IndexingOnline, Cindex, for instance) only slightly rely on the analysis of the document content.

Arguing that computational terminology is now able to give further assistance, we have designed a new indexing method, which exploits terminological tools to facilitate the indexing task. From the analysis of the document text, our IndDoc system automatically builds an index draft that is then validated by an indexer through a dedicated interface. The resulting index is a terminological network which nodes correspond to the index entries associated with page numbers.

When developing such an innovative method, which cannot be directly compared with existing ones, one has to think of how it can be evaluated. The general approach must be validated as soon as possible, *i.e.* without waiting that a user set can test an operational system. This paper addresses this preliminary evaluation problem. Even if the indexing task remains difficult to evaluate, we describe the method that we designed to

nevertheless assess the quality of our approach towards indexing.

The first two sections describe what is a back-of-the-book index and present the IndDoc overall method and architecture. The section 4 explains the evaluation difficulties that one has to face when evaluating indexing tools. Our evaluation protocol is respectively presented and discussed in section 5.

## 2. Back-of-the book indexes

Traditionally, the index that is placed at the back of a book or document is an alphabetic list of descriptors associated with page numbers or page ranges. It is composed of two parts: a nomenclature and a list of references (see Figure 1).

The nomenclature is a list of descriptors, the index entries, that give access to the document content. Some index nomenclatures are structured and present explicit semantic relations between descriptors. This structure is usually mainly hierarchical. The specific descriptors are presented as sub-entries of entries that correspond to more generic descriptors (see *knowledge* and *knowledge representation* on Figure 1). Some indexes also have synonymy relations, variations (the expanded form of *AI*) or more generally association links (often called *see* or *see also*).

An index is therefore composed of a terminological network (the nomenclature made of descriptors and terminological relations) and we developed a tool to automatically produce a book index out of the document terminological analysis.

Acquisition	7, 21, 78-81, 250
AI see Artificial Intelligence	
Artificial Intelligence	43, 97, 134
Knowledge	26-32, 76-77, 89, 211-215, 228
Acquisition (see also Acquisition)	228
Representation	25-29, 80, 132-136, 250
<b>Nomenclature</b> <span style="margin-left: 200px;"><b>References</b></span>	

Figure 1: Index example

### 3. IndDoc indexing method and architecture

Our indexing approach is based on the last decade results in computational terminology and more generally in natural language processing. The architecture of our system is presented on Figure 2.

The terminological analysis is itself composed of two steps, the term and relation extraction respectively. For the experiment reported here, we exploited Lexter (Bourrigaut et al. 1996) and Syntex (Bourrigault & Fabre, 2000) for extracting terms and we developed our own relation extraction module, which combines some contextual extraction patterns extraction (Hearst, 1992, Morin, 1999, Charniak & Berland, 1999), the syntactic analysis of the terms and the projection of a synonymy dictionary (Hamon & Nazarenko, 2001).

Once a draft of the terminological network is built, IndDoc looks for the term occurrences in

order to connect the nomenclature entries with the document segments (reference calculus). The next procedure aims at ranking by relevance order both the list of terms and the set of reference segments for each index entry. This ranking procedure is important, for instance, to adjust the index length to fit the editorial constraints. It also guides the validation process. This ranking is based on the frequency of terms and their repartition over the document but also on cohesion and salience factors (typographical, lexical and textual), which establish the relative importance of index descriptors and document segments as reference candidates (Aït El Mekki & Nazarenko, 2005).

The resulting index, however, is only a draft index, since all the extracted terms and relations are not well formed or relevant for a given index. An experienced indexer must manually validate the result. We developed a dedicated interface to help this validation process.

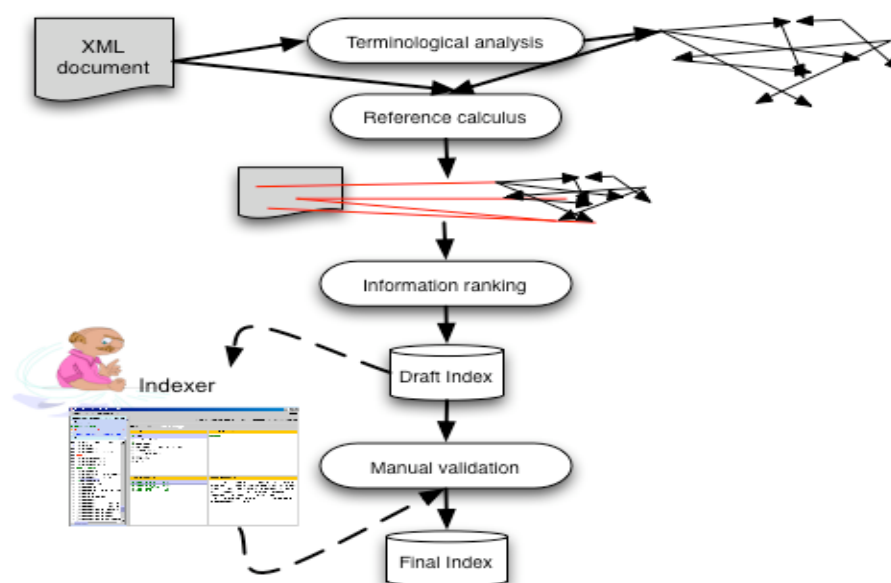


Figure 2: IndDoc architecture

### 4. Evaluation problem

When developing such a system, it is extremely important to be able to evaluate it. The goal is twofold:

- Assess the adequacy of the overall approach
- Evaluate the usefulness of the terminological tools not *per se* as it is performed in evaluation campaigns<sup>1</sup> and extractor comparison (Cabre *et al.*, 2001) but in the specific application context of index building.

However, evaluating IndDoc results raises two separate questions that are traditional in terminological processing. Since our indexing method is a cooperative one, it is difficult to

evaluate the specific contribution of the automatic tool. It is also difficult to evaluate the quality of indexes since there is no objective reference. Two indexers do not produce the same index for a given document. The indexing guides only give general recommendations like: "Try to be as objective as possible in the choice of entries and include those entries in the index that you think a reader may want to look up. Refer only to pages where an item is discussed, not just mentioned." (Mulvany 1993). More generally it is acknowledged that indexers lack of systematic evaluation protocols (Wyman, 2005).

An additional problem comes from the fact that IndDoc is still a laboratory prototype, which cannot be easily tested by a group of users in realistic working conditions. As any system developer, we nevertheless need early evaluation elements to

<sup>1</sup> See for instance the CESART campaign: <http://www.elda.org/article137.html>.

decide whether to pursue the development or to abandon it.

## 5. Elements of evaluation

Our indexing method should target two types of users: the indexer who builds a source index out of a draft index using a validation interface, and the reader who uses the resulting index for information localisation. However, we consider that the indexer is responsible for the adaptation of the index to the expected reader's profile. In this paper, we only evaluate the impact of the automatic indexing process on the cost and quality of the indexer's task.

The hypothesis underlying the IndDoc system development was that terminological processing would enable indexers to build richer indexes more

easily than with traditional indexing tools. Really validating the above hypothesis, however, would require to have indexers testing the IndDoc system in a more systematic way and to analyze their feedback. Such a large-scale experiment cannot be set up from scratch. We need a preliminary evaluation beforehand. This is the goal of the elements of evaluation that we described here.

To get an idea of the quality of our indexing method, we compared several indexes produced for the same documents. We deliberately re-indexed documents which had been previously been published with an index. We made three types of comparisons (see the numbered bold arrows on Figure 3).

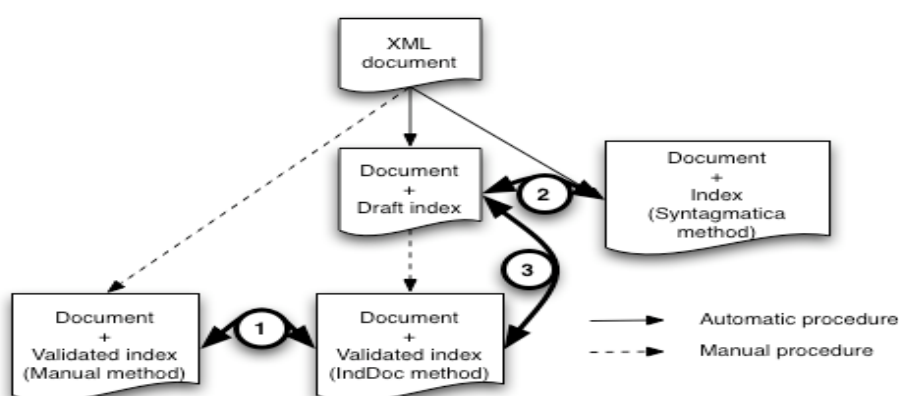


Figure 3: Schema of the evaluation protocol

In order to evaluate the added-value for an indexer to exploit an indexing tool such as IndDoc, we compare traditional indexes (traditional manual indexing) and IndDoc indexes (automatic indexing and manual validation) for the same documents<sup>2</sup>.

We also compare the draft indexes produced by IndDoc and the equivalent draft indexes produced by existing indexing tools such as Syntactica<sup>3</sup>, which analyses the text of the document and proposes every noun phrases as index descriptors. This aims at assessing the contribution of terminological analysis and information ranking to indexing.

We finally compare the draft indexes automatically built by IndDoc and the final indexes resulting from the indexers' validation. This

comparison helps to evaluate the quality of the automatic process of IndDoc.

The experiments reported here have been performed on three different corpora, mainly focused on linguistics (LI), Artificial Intelligence (AI) and Knowledge Acquisition (KA) and the results are presented on Table 1. These figures are globally encouraging. They show that the IndDoc procedure produces much richer indexes than the traditional author's manual indexing. The size of the index considered as the number of descriptors and the proportion of relations per descriptor is significantly increased.

The second set of comparisons shows that IndDoc outperforms existing tools because it proposes some relations between descriptors and it filters out the descriptor lists (we estimate that Syntactica would produce 10 000 descriptors for KA whereas the editors who produced the final index out of IndDoc results refused to validate more than 2 000 ones).

The third type of comparison brings out contrasted results. The precision of the relation extraction is rather good (more than 65%, even though the method need to be improved) much better than the precision rates of descriptor extraction. This last result does not take the ranking

<sup>2</sup> In the reported experiments, the traditional indexes are those with which the books have been published. A different person from the original indexers, which, in this case, were the document authors, has validated the IndDoc draft indexes.

<sup>3</sup> The Syntactica analysis has been simulated, since Syntactica only processes English documents whereas our first IndDoc experiments were done on French documents.



into account, however. The ranked precision rates, which reflect the capacity of the system to top rank

good descriptors, are much better (more than 75 %) and encouraging.

	Monographs		Collection
	LI	AI	KA
Corpus size (# of words occurrences)	42 260	111 371	122 229
Existence of an original manual index	Yes	Yes	No
Existence of a draft index	Yes	Yes	Yes
Existence of an IndDoc index	Yes	Yes	Yes
Precision of descriptor extraction – comparison 3	33%	44%	71%
Ranked precision of descriptor extraction – comparison 3	77%	83%	83%
Precision of relation extraction – comparison 3	65 %	71 %	80%
Size increase (# of descriptors) – comparison 1	+85%	+50%	Non applicable
Size increase (average # of relations per descriptor) – comparison1	+166%	+300%	Non applicable

Table 1: Evaluation results for three different corpora. The precision figures give the proportion of relevant information in the draft index. The percentage figures show that IndDoc index is much richer than the original published indexes (the book authors acknowledged the overall quality of these large indexes).

## 6. Conclusion

Even if evaluating terminological products is known as a difficult task (man-machine cooperation, subjectivity of the quality criteria and heterogeneity of the terminological methods and goals), we showed that it is possible to evaluate the contribution of terminological tools such as term and terminological relation extraction in the context of a given application (here, back of the book indexes). This type of evaluation procedure is relatively easy to set up compared with user-based ones. It does not support a definitive assessment but it gives useful indications of the method quality prior to large experimental evaluations.

## 7. References

Aït El Mekki, T., A. Nazarenko. 2005. "Using NLP to build the hypertextual network of a back-of-the-book index." In *Proc. of the Recent Advances in Natural Language Processing (RANLP)*, Borovets, Bulgaria.

Berland, M. and E. Charniak. 1999. "Finding parts in very large corpora." In *Proc. of the 37th Annual Meeting of the Assoc. for Computational Linguistics*, pp. 57-64. New Brunswick NJ.

Bourigault, D., I. Gonzalez-Mulllliez and C. Gros. 1996. "LEXTER, a natural language processing tool for terminology extraction." In *Proceedings of the 7th Int. Congress EURALEX*, pp. 771-779. Göteborg, Sweden.

Bourigault, D. and C. Fabre. 2000. "Approche linguistique pour l'analyse syntaxique de corpus." *Cahiers de Grammaire* 25, pp. 131-151.

Cabré, M. T., R. Estopá and J. Vivaldi Palatresi. 2001. "Automatic term detection: A review of current systems". In *Recent Advances in Computational Terminology*, Bourigault, D., C. Jacquemin and M.C. L'Homme (eds.), pp. 53-87, John Benjamins.

Hamon, T. and A. Nazarenko. 2001. "Detection of synonymy links between terms: experiment and results." In Bourigault D., C. Jacquemin and M.C. L'Homme (eds.). *Recent advances in*

*Computational Terminology*, pp. 185-208, John Benjamins.

Hearst M. A. 1992. "Automatic acquisition of hyponyms from large text corpora." In *Proc. of the 15th Int. Conf. on Computational Linguistics*, pp.539-545. Nantes, France.

Indexing Research. 2000. Logiciel Cindex. <http://www.indexingonline.com/index.php>.

Jacquemin, C. 1999. "Syntagmatic and paradigmatic representations of term variation." In *Proc. of the 37th Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pp. 341-348. Univ. of Maryland, Maryland.

LangPower Computing. 2004. Logiciel Indexing online. <http://www.indexingonline.com/>.

Lin, D. 1998. "Automatic retrieval and clustering of similar words." In *Proc. of Int. Conf. on Computational Linguistics (COLING/ACL)*, pp. 768-774. Montréal, Canada.

Macrex. 2003. Logiciel Macrex. <http://www.macrex.cix.co.uk/>.

Morin, E. and C. Jacquemin. 1999. "Projecting corpus-based semantic links on a thesaurus." In *Proc. of the 37th Annual Meeting of the Assoc. for Computational Linguistics*, pp. 389-396. Maryland.

Mulvany, N.C. 1993. *Indexing Books* (Chicago Guides to Writing, Editing & Publishing). Chicago, USA: The University of Chicago Press.

Nazarenko, A., Hamon T. (eds). 2002. "Structuration de terminologie." *Traitement Automatique des Langues*, 43(1), pp. 7-18.

University of Chicago Press Staff (ed). 2003. *The Chicago Manual of Style*. Chicago, USA: University of Chicago press.

Virginia Systems. 2004. Logiciel Sonar BookEnds. <http://www.virginiashystems.com/>.

Wacholder, N., Nevill-Manning, C. 2001. "Workshop report: The technology of browsing applications, Workshop held in conjunction with JCDL 2001." *SIGIR Forum* 35(1), pp. 6-19.

Wyman L.P. 2005. "Judging Indexes", In *"A to Z" the bulletin of the STC's Indexing SIG*.

# Semi-Automatic Checking of Terminographic Definitions

Selja Seppälä

Terminology, Multilingual Information Processing Department (TIM)  
School of Translation and Interpretation (ETI), University of Geneva  
Bd du Pont-d'Arve 40, CH-1211 Genève 4  
e-mail: selja.seppala@eti.unige.ch

## Abstract

Terminographers are expected to conform to a set of definition writing rules and principles, and to specific lexicographical traditions, all of which entail the absence or presence of a number of lexico-syntactic markers. Since terminographic definitions are a synthesis of several specialised defining contexts which may contain these markers, terminographers might produce poor quality definitions. In this paper, we present a method for semi-automatically checking definitions. We draw up a typology of common errors in definitions and examine their distribution in a corpus. We show that potentially half of the errors can be automatically detected, among which almost 30% with simple spell checkers. We then outline a method for the implementation of a semi-automatic definition checker and its drawbacks. Finally, we evaluate in more detail a simple method, based on pattern extraction, which potentially covers around 20% of the errors. The experiment shows that the major issue of this method is finding the best possible markers according to one's definition writing options, and that the extraction patterns must be refined to reach the best possible results.

## 1. Introduction

Writing terminological dictionaries requires defining concepts in specialised domains. The definitions should be as consistent as possible for the product to meet certain standards. (ISO, 2000) Terminographers are therefore expected to conform to the generally accepted definition writing rules and principles, and to specific lexicographical traditions, which entail the absence or presence of a number of lexico-syntactic markers. Since terminographic definitions are a written synthesis of several defining contexts (L'Homme, 2004; Meyer, 2001) extracted from specialised references, and since these markers may appear in the defining contexts, terminographers, who do not always conform to some of these general principles, may leave in such markers, thus producing poor quality definitions. The markers can therefore be used to detect poor definitions. Our aim is to provide terminographers with a method for semi-automatically checking definitions. In this paper, we present a typology of common errors in definition writing, a study to determine to what extent these can be automatically corrected, or at least located, and an overall method to automate the checking of definitions. The difficulties raised by this kind of global task are outlined. Finally, a short evaluation of a simple automatic method to detect a part of these errors is discussed and some solutions to achieve better performance are proposed.

## 2. A Typology of Errors

According to previous work (Seppälä, 2004) and from the observation of a corpus of student definitions, errors found in terminographic definitions can be classified into five general categories, divided into a total of 15 subcategories, as follows:

1. **Basic linguistic errors**: spelling mistakes and grammatical problems.
2. Errors due to the **non-observation of lexicographic traditions**, i.e., in French:
  - (a) a definition **neither beginning with a capital letter nor finishing with a full stop** or

- (b) a definition **beginning with an article** (definite or indefinite).

3. Errors due to the **non-observation of definition writing rules and principles** (See for example Auger & Rousseau, 1988; Dubuc, 1978; ISO, 2000; Rondeau, 1984):

- (a) use of more than one sentence and, thus, of **punctuation** other than commas, i.e. *full stops, brackets, colons, semi-colons*, etc.;
- (b) including **the definition of one of the terms in the definition**, which can be seen through expressions like *namely, i.e., that is to say*, etc.;
- (c) **specifying the domain** to which the defined concept belongs with expressions like *In...*;
- (d) use of indexical expressions such as *we, your, here, today*, etc., or locutions like *etc., generally, for example, according to*, which are **not objective and generalising enough**;
- (e) use of **redundant generic** elements, such as *All the..., Every..., A type of...*;
- (f) **inconsistency in number and grammatical category** between the *genus* and the (main) term;
- (g) **not using the *genus proximus*** but some more general generic term (situated further up in the hierarchy);
- (h) having (one of) the **term(s)** designating the defined concept **in the definition**.

4. **Stylistic and formulation errors**.

5. **Content errors**, i.e. relative to the concept defined, such as:

- (a) use of a **conceptually inappropriate generic** term;
- (b) use of **non-defining encyclopaedic information** that should preferably appear in another field of the terminological record;
- (c) **others**, such as an incomplete definition, defining the wrong concept or including the definition of another concept instead of the referring term.

### 3. Error Distribution in Terminographic Definitions

To determine how common errors are distributed, we have studied a corpus of terminographic definitions. Each error (previously annotated for teaching purposes) is marked and categorised according to the typology presented in section 2. The presented tables (tables 1 and 2) were obtained after correction of some originally missing errors found in the corpus during testing (see section 7.1).

Whenever the same part of a definition accumulates two or more types of errors, each occurrence is counted once. Each type of error can also appear several times in one definition, and is counted as many times as it appears.

#### 3.1. The Corpus

The corpus used in this study consists of 223 terminographic definitions in French, extracted from 22 short terminological glossaries written by students specialising in terminology. Qualitatively speaking, most glossaries were considered as “acceptable” (15 were graded 4-5 out of 6<sup>1</sup>, three as “poor” (< 4) and four as “good” (> 5). The corpus can be considered as representative of what one could expect, for example, from trainees in terminography or from translators having only some basic knowledge of definition writing.

#### 3.2. Error Distribution

Table 1 shows the error distribution for each type of error.

ERROR TYPE		nb.	%	%
1.	spelling & grammar	50	16.2	28.2
2a.	main sentence punctuation	37	12	
2b.	articles at the beginning	1	0.3	24.6
3a.	not one sentence & other punctuation	34	11	
3b.	other definition in definition	6	1.9	
3c.	domain in definition	0	0	
3d.	non objective/general	21	6.8	
3e.	redundant generic	6	1.9	
3f.	generic: inconsistent nb./gram. cat.	6	1.9	
3g.	non <i>genus proximus</i>	1	0.3	
3h.	term in definition	1	0.3	
4.	style & formulation	55	17.8	
5a.	inappropriate choice of generic	36	11.7	
5b.	non-defining information	14	4.5	
5c.	other content errors	41	13.3	
TOTAL		309	100	100

Table 1: Error distribution

Errors of type 1 and 2a, which can be considered as spelling errors, since the French lexicographic tradition conforms to the standard definition writing rules for a

<sup>1</sup> Marks range from 1 to 6, where 6 is the maximum.

sentence, sum up to 28.2%, i.e. more than a quarter of the total. Errors due to “infringements” of the definition writing rules and principles (3a-h), which in fact also include type 2b, represent 24.6%. Finally, errors related to human judgment and conceptual factors (types 4 and 5a-c) represent around half of the errors with 47.2%.

### 4. How Much Can Be Automatically Checked?

This table tends to confirm that it may be profitable to develop a method for automatically checking terminographic definitions, since on average every definition has 1.4 errors in it (309 errors in 223 definitions). But how much can be automatically checked? Some of these errors do not seem to be suited for automatic detection, because they are more a matter of content and personal judgement. This is the case for types 4 and 5a-c (47.2%), which should therefore be manually located. However, automating the detection of type 1 to 3e errors (50.2% of the total) seems feasible, as we will see in this paper, since these are more related to formal aspects of the definition. The same applies to type 3f-h errors (2.6%), provided some more complex pre-processing of the data is performed, such as morpho-syntactic annotation or a proper segmentation of the generic elements. As such, about half of the errors in definitions can be automatically addressed.

Numerically speaking, the potential error coverage of an automatic definition checker would thus be 52.8%, as shown in table 2. Knowing that type 1 and 2a errors (28.2%) can already be (semi-)automatically corrected with current spelling and grammar checkers (within the known performance of these tools), we focus on the resolution of the remaining 24.6%, and particularly, in this paper, of type 2b and 3a-e errors (22%), which do not require the use of complex procedures.

CHECKING METHOD	%	%	%
automatic with existing tools (1-2a)	28.2	50.2	52.8
automatic with method presented below (2b-3e)	22		
automatic with pre-processing (3f-3h)	2.6	2.6	
manual (4-5c)	47.2	47.2	47.2
TOTAL	100	100	100

Table 2: Potential coverage of checking methods

### 5. Outline of a Semi-Automatic Definition Checker

As already mentioned, our procedure is based on the automatic detection of a number of lexico-syntactic markers indicating errors due to the non-observation of definition writing rules and principles.

However, this simple method does not cover all of the errors that can be automatically located. Therefore, an automatic definition checker should also integrate other types of implementations according to the type of error: on the one hand, a classical spelling and grammar checker; on the other, a more complex method that includes different kinds of pre-processing of the data.

For instance, the detection of most of the errors related to the generic element (3f and 3g) is subject to the proper segmentation of the element, which requires a thorough study of the nature of the *genus* and the development of adequate routines. The checking of errors implying (one of) the term(s) (3f and 3h) is subject to the previous detection and/or grammatical annotation of both the term(s) and the definition. As for the detection of a generic element which is not the closest one in the corresponding hierarchy (3g), the difficulties are even greater: it requires a formalised hierarchy of the domain to which the defined concept pertains, as well as a proper segmentation of the generic element.

But is it worth implementing such costly pre-processing methods to detect errors which cover only around 3% of the total errors? It is not our purpose to answer this question here; we will instead present a simpler method to cope with potentially around 20% of the errors.

## 6. A Simple Checking Method

The proposed method is based on a Perl script performing pattern matching of markers in definitions expressed in the form of regular expressions. It covers errors of types 2b and 3a-e. The central challenge of this method is therefore finding the best possible markers (*il s'agit de/d' ≈ that is*) and patterns ('*il s\ 'agit d[e|\ ' ]'*).

The lists of lexico-syntactic markers were first made *a priori*, using introspection, on the basis of the commonly accepted definition writing rules and principles, and of the errors generally observed in student's glossaries. The patterns were then evaluated against the above-mentioned corpus. Finally, some adjustments were made during testing in order to achieve better performance. We present the results and adjustments for three successive tests.

### 6.1. The Markers

The markers below were the ones used for the first test. Some of them were modified or suppressed, and other ones were added, depending on their initial performance. The most significant ones are discussed in section 7.

The markers can be grouped according to location constraints inside the definition: some of them must be searched for at the beginning of the sentence, whereas others may be sought in any position. The distinction between upper and lower case is ignored.

#### 6.1.1. At the Beginning of the Definition

- To locate type 2b errors, i.e. those due to the non-observation of the French lexicographic tradition, since they begin with a definite or an indefinite article, the program looks for articles *l', le, la, les, un, une* and *des* at the beginning of the definition.
- Two markers are used to detect specification of the domain to which the defined concept belongs: *en* and *dans le domaine de*.
- To spot uses of redundant generic elements, we search for the following markers: *tout(e), type de, sorte de, on entend par, il s'agit de, (c'est un(e))*.

#### 6.1.2. Within the Definition

- To detect definitions of other concepts inside a definition, we search for markers introducing a definition: *c'est-à-dire, on entend par, il s'agit de, à savoir, signifie, désigne, désignant, s'appelle, s'appelant, soit*.
- To find definitions exceeding one sentence and including unconventional punctuation we look for possible occurrences of *full stops inside the sentence, brackets, colons, semi-colons, exclamation and question marks, square brackets and hyphens*.
- To locate elements opposed to the generalising and objective nature of a definition, we look for four types of markers: markers expressing generality or specificity (*ou, etc., en règle générale, généralement, habituellement, particulièrement, en particulier, exemple, souvent, parfois*), local and temporal indexicals (*à présent, aujourd'hui, à l'heure actuelle, ici, voici*), causative markers (*car, puisque, donc*), and markers referring to a "speaker" (*selon, d'après, je, tu, me, te, nous, vous, voilà, certes*). The latter were included, even if it is highly unlikely that, for example, personal pronouns, particularly *tu* or *vous* (*you*), appear in definitions, since specialised texts from which the defining contexts are extracted seldom if ever contain them.

## 7. Evaluation of the Simple Method

### 7.1. Results

As mentioned before, the tests were done by successively adjusting the markers and the extraction patterns in order to achieve the best possible performance with regard to the test corpus.

The first evaluation was made with a set of 59 patterns that was created by hand without reference to a particular corpus. The second evaluation was made after modification of the markers to make them conform more to the correction criteria adopted in the correction of the corpus. Nine markers were added and four suppressed. The third evaluation was made after refinement of some "noisy" extraction patterns. The total number of errors, of the types considered here (2b-3e), marked by hand in the entire corpus was 58.

type of error	# of errors in corpus	1 <sup>st</sup> test		2 <sup>nd</sup> test		3 <sup>rd</sup> test	
		R	P	R	P	R	P
2b	1	1	1	1	1	1	1
3a	34	.97	.97	1	.97	1	1
3b	4	.25	.5	1	.57	1	.57
3c	0	1	1	1	1	1	1
3d	13	.38	.4	1	.56	1	.59
3e	6	.83	1	.83	.1	.83	1
<b>TOTAL</b>	<b>58</b>	<b>.77</b>	<b>.25</b>	<b>.98</b>	<b>.80</b>	<b>.98</b>	<b>.83</b>

Table 3: Precision and recall measures for each test

Table 3 shows that when the set of markers is matched to the correction criteria used for correcting the corpus (i.e. less strict criteria), recall rises considerably. On the other hand, while having the right markers, the adjustments of the corresponding extraction patterns made between the 2<sup>nd</sup> and the 3<sup>rd</sup> run improved precision. Therefore, we can say that recall seems more significant to measure the adequateness of the markers with regard to given correction principles and that, once the markers comply with one's expectations, precision becomes more important.

%	precision	corrected precision
1 <sup>st</sup> test	25	28
2 <sup>nd</sup> test	80	92
3 <sup>rd</sup> test	83	94

Table 4: Actual error detection rate

Moreover, the results also show that if extremely strict writing rules are applied, as it was the case in the first set of markers, this simple method detects more errors than manual detection<sup>2</sup>. Indeed, part of the noise could have been considered as actual errors if the manual correction had been equally strict and had listed them. However, even considering a less strict set of markers, which in fact corresponds to the one used in the corpus, this method may detect more systematically each type of error. Refinement of extraction patterns indeed shows an increase in the number of errors automatically detected but not marked in the corpus, although they should have been (shown in table 4). This implies that this checking method is more systematic and, in a way, more "reliable".

The choice of markers should be parametrisable in order to obtain a better recall. As for patterns, these should be adequately conceived to raise precision. Therefore, to make the method more effective, the sets of markers and corresponding patterns might also be automatically learned from a sample corpus corrected following chosen correction criteria, which can be more or less strict. This would produce markers that are compliant with the expected definition writing options and that do not contain unproductive elements (like markers that would hardly be found in this kind of text and, thus, slow down the processing) and better adapted patterns. However, automatic learning of markers requires separate training and test data sets.

Since an actual parametrisable tool would allow a choice of the markers, thus letting the writer decide whether he/she wants noise in the corrections, the important measure to use to evaluate the system is precision (knowing that recall is constant).

## 7.2. Difficulties and Solutions

The difficulties affecting the performance of the method at each step of the successive evaluations can be classified into four major categories.

### 7.2.1. Spelling Mistakes

Some of the errors were not found because of spelling mistakes in the corpus which made it impossible for the patterns to match the erroneous strings. This suggests that the definitions should be previously checked for spelling mistakes.

### 7.2.2. Divergences in the Definition of "Error"

Considerable noise was present due to the fact that, in the test corpus, definition writing rules were not applied as strictly as we expected them to be. Generally speaking, these problems may arise whenever the terminographer's definition writing options are divergent from the canonical ones, or at least from some pre-established standards (such as corporate requirements). Two examples are examined:

- First, we can take the case of the causative markers *car* and *donc* (*because* and *therefore*), which seem to be relevant markers to spot errors since they convey the personal judgment of the writer (Bailly & Toro, 1971) and thus should not appear in a definition. A definition should be written in an impersonal form, the subject being the concept defined, not the author of the definition. However, the occurrences of these markers were not considered as errors in the reference corpus<sup>3</sup>, thus entailing some divergence between the results of the manual and the automated (1 *car* and 2 *donc* found) correction methods.
- Another example is the use of adverbial phrases or adverbs like *généralement* or *souvent* (*generally* or *often*). Strictly speaking, definitions should not include these, either because they are not generalising enough, and thus non-defining, or because they are "redundant" with regard to the nature and function of the definition: they make generalisations of the common characteristics of the defined objects. Specifying that the characteristic is to be considered on a general basis appears thus superfluous and redundant. It may also suggest that the information is non-defining, and should therefore be rejected or placed in another field of the terminological record. However, according to a prototypical view of the terminographic definition (Temmerman, 2000), one would probably not want to suppress this specification and, as such, not consider it as an error.

Here, the problems are not inherent to the error detection method, but to the terminographic choices made by the author of the definitions, i.e. to the definition of an "error". Therefore, the solution would be to implement the detection of these "errors", leaving them optional and/or prompting some reformulation tips, in order to make the defining sentence conform to the different defining options.

### 7.2.3. Ambiguity of Markers

Some of the markers used in the first test produced a considerable amount of noise due to the fact that they

<sup>2</sup> See explanation for the *ou* marker in part 7.2.3.

<sup>3</sup> Even if they would have been better formulated with *du fait que* instead of *car*, or *de ce fait* instead of *donc*.

were ambiguous. Two types of ambiguity were discovered:

- Ambiguity in the kind of information extracted. An example is the case of the conjunction *ou* (*or*), which was included in the set of markers because a definition should assign a limit and avoid enumerations of particular variants of the defined concept, as in “*Culture* *d’arbres-abris, ou arbrisseaux ou autres végétaux*, destinée à protéger de l’insolation, du froid, du vent, etc. les jeunes plants d’une plantation.”<sup>4</sup>. In this example, the information inside the box could be replaced by a more general term, or even suppressed, since it is already included in the concept of *culture*. However, the results show that this marker also indicates an alternative between a limited set of things (generally two), such as in “*Concentration anormalement élevée de constituants gazeux, liquides ou solides dans l’atmosphère.*”<sup>5</sup>. In this case, it does not compromise the definiteness of the definition. Since none of the 2b-3e errors marked in the corpus contained this marker, we decided to suppress the marker from subsequent tests. However, if one wants to check the appropriateness of each and every occurrence of this marker, it should be left optional.
- Another type of ambiguity is ambiguity in the meaning of the marker, as in the case of *soit* (*that is to say, either or be*<sup>6</sup>). This marker was originally included in the set of markers used to locate definitions of other terms in a definition. However, among the 17 occurrences automatically detected, 14 meant “*either*”, 1 corresponded to the verbal form, and only 2 were actual definition markers, which had been marked as type 5 errors. The “noise” due to the extraction of “*soit*” meaning “*either*” could be avoided by constraining the extraction pattern so that it would discard as non-erroneous all the cases where the marker appears exactly two times and is, each time, preceded by a comma. But this would probably also discard actual errors.

In order to avoid some of these ambiguities and achieve more generalised patterns, the simple method could be tested on previously morpho-syntactically tagged definitions. The difficulties would then probably arise from pre-processing errors, which would affect the processing of subsequent data, thus reducing the performance of the method. However, since performances achieved without any pre-processing can be considered as good (provided the method is still tested on other corpora), it seems too cost-inefficient to implement pre-processing to cope with only a few problematic cases.

#### 7.2.4. Inadequate Extraction Patterns

Some extraction patterns appear to be too productive or not productive enough. This was for example the case

of the pattern initially assigned to match parenthesis, which also detected parenthesis indicating a plural. Since such parentheses are not considered erroneous, the extraction pattern was modified accordingly to ignore these cases. Sometimes, even adding a simple space before and/or after a lexical pattern was enough to avoid getting unwanted matches. Such was the case when adding a space before and after “*ici*”. In fact, our pattern takes into account all the occurrences of *ici* (*here*) preceded or followed by a comma or a space: “[\,| ]ici[\,| ]”.

#### 7.2.5. Synthesis

The successive tests using this simple method show that the level of performance depends mainly on three factors:

- The correctness of the tested data.
- The relevance of the markers with regard to determined (variable) definition writing options.
- The adequacy of the extraction patterns with respect to the significant markers.

Besides running a spellchecker beforehand and refining the patterns, the easiest way to cope with these problems and to meet various types of definition writing requirements, would therefore be to implement the method in an adaptable way, by adding parametrisation of checking options and/or prompting with correction (reformulation) tips, such as “*In case of an enumeration, replace it by the class (the genus) grouping all of its elements. For example: replace “cat, dog, mouse, etc.” by “animals”.*” when detecting “*etc.*” or an “*or*” in an enumeration.

## 8. Conclusion

In this paper, we have seen that errors occurring in terminographic definitions can be classified according to a five point typology, with a total of 15 examined subcategories, and that half of them can be semi-automatically corrected, providing some amount of pre-processing of the data. We have outlined a method for the implementation of a semi-automatic definition checker and its drawbacks. We have then evaluated in more detail a simple method, based on pattern extraction, which potentially covers around 20% of the errors.

The experiment shows that the major issue of this method is finding the best possible markers according to one’s definition writing options, and that the extraction patterns must be refined in order to reach the best possible results. Therefore, this simple definition checking method should also be usable for automatically extracted or generated definitions, providing adequate adjustments in the markers and the patterns are made.

Even though the proposed method should be further tested, we believe that it is a valuable tool to help those who write definitions. Its major advantage is that it is easy to implement, since it needs no complex pre-processing of the data, except spell checking. It should also be easy to adapt to other languages by creating new sets of markers and the corresponding extraction patterns. An actual application should be parametrisable and could integrate some extended functions like optionality of definition writing “rules” (i.e. markers) or prompting with correction tips. Combined with classical grammar and spell checkers, this semi-automatic definition checking method would be

<sup>4</sup> “*Planting of a crop of trees or shrubs or other plants...*”

<sup>5</sup> “*Abnormally high concentration of gaseous, liquid or solid constituents in the atmosphere.*”

<sup>6</sup> Third person of the subjunctive of the verb *être* (*to be*).

suited for a “quick” formal checking of definitions, though it would not replace more thorough examination of their content.

## 9. References

- Auger, P. & Rousseau, L.-J. (1988). *Méthodologie de la recherche terminologique*. Québec: Office de la langue française.
- Bailly, R. & Toro, M. d. (1971). *Dictionnaire des synonymes de la langue française*. Paris: Larousse.
- Dubuc, R. (1978). *Manuel pratique de terminologie*. Montréal, Paris: Linguatex, CILF.
- ISO (2000). *Travaux terminologiques : principes et méthodes (ISO 704)*. Genève: ISO.
- L'Homme, M.-C. (2004). *La terminologie : principes et techniques*. Montréal: Presses de l'université de Montréal.
- Rondeau, G. (1984). *Introduction à la terminologie*. Québec: Gaëtan Morin.
- Seppälä, S. (2004). *Composition et formalisation conceptuelles de la définition terminographique*. Mémoire de DEA en traitement informatique multilingue. Université de Genève, École de traduction et d'interprétation.
- Temmerman, R. (2000). *Towards new ways of terminology description : the sociocognitive-approach*. Amsterdam, Philadelphia: John Benjamins.