Concord: A Tool for the Analysis and Concordances of the Terminological Constituents

Plini Paolo¹, Mastidoro Nicola², Mazzocchi Fulvio¹

Abstract

Concord, a software for the analysis of the terminological constituents and for the creation of concordances, has been developed within a cooperation between CNR and Êulogos. It produces a term index that represents an evolution of the traditional KWIC indexes. It could support the activities of development of terminological databases and in particular thesauri. The main feature of Concord is the capability to “learn” from the already performed analysis, making suggestions to the user for further actions. The index generated could be exported in order to be uploaded in different applications.

1. Introduction

Thesauri are vocabularies of controlled language providing a semantic organization of terminology. Traditionally they are presented according to a systematic or alphabetic order, and they can include a permuted list of the terms. One step of the editing process of thesauri can be represented by the compilation of the so called KWIC (KeyWord In Context) concordance. The creation of this index system produces a permuted list where each keyword is alphabetically ordered with a certain amount of other words presented before and after.

In thesauri the KWIC is applied to terms so that a particular word is listed in the middle column, with a certain amount of text on either side. One limit of KWIC is that it is unable to handle complex terms, like “bioengineering” where a subset of the term is itself relevant from the point of view of indexing and retrieval.

2. The operational context

During the development of EARTH - Environmental Applications Reference Thesaurus (Mazzocchi, Plini 2005) and of STeRNA, the Italian National Environmental Reference Terminology System, this problem was approached first theoretically and then through the creation of a specific tool.

Several problems arise when a term is analysed from a morphological point of view aiming at indexing its terminologically relevant elements. For example “bio” has to be considered as a prefix of “diversity” in the term “biodiversity”, but it does not represent a prefix in the term “biota”. In the same way, the term “biodiversity” has to be split in two parts that will be represented in the index but at the same time the whole term must be kept and represented.

There is not the possibility to define and follow a common procedure, since each case has to be evaluated both from the morphological and from the terminological point of view.

¹ CNR-IIA-EKOLab, Environmental Knowledge Organisation Laboratory. Via Salaria, km 29,300. 00016 Monterotondo stazione (RM), Italy – http://uta.iia.cnr.it/ – {plini; mazzocchi}@iia.cnr.it
² Êulogos SpA -Via Cimone, 59 - 00141 Roma - http://www.eulogos.it/ - n.mastidoro@eulogos.net
Starting from this considerations, EKOLab activated an interaction with Ëulogos in order to develop a tool capable of ease the work, flexible and customisable.

The EKOLab-Êulogos working group defined a set of basic features and constraints:
- Handling more languages;
- Possibility to mark a term as an atom, which means that the term cannot be split;
- Possibility to accept as an atom a part of the term and to discard the other part as a stop word;
- Capability to split a term into its components keeping the term itself as an entry;
- Recording of the performed actions finalised to the suggestion of future actions;
- Identification of stop words within terms according to recorded knowledge;
- Possibility to ignore the part of the string closed into brackets;
- Capability to store not allowed terminations;
- Capability to store stop words;
- Dynamic browsing of terms and term elements as concordance of terms in thesaurus;
- Generation of a multi-level index (terms + defined term elements) in a standard file;
- Import-export functions basically in TXT and XML format;
- Possibility to update the database or to merge additional parts without loosing the already performed actions;
- Printing functions.

3. Software details

In order to meet the project requirements, Concord has been conceived and developed in the framework of Êulogos’ language engineering environment.

From a terminological and linguistic point of view, Concord is language-independent and can be applied to any Latin-characters language. Special characters are fully supported. Two interface languages are available: Italian and English. An easy-to-use interface has been developed according to the terminologist’s point of view.

The linguistic engine of Concord manages atoms, stop words and complex terms according to a self-learning logic, allowing the system to apply on new terms each learned element and structure.

For each term, Concord proposes a pre-assigned representation of its elements. The user reviews the proposal and can modify it in each element. All terms and elements are browseable as concordances.

The concordance method of Concord has been derived from the IntraText Digital Library (Mastidoro 2004).

3.1. The working flow

A Concord project can be seen as divided into three elaboration steps:
1. Analysis of terms;
2. Analysis of atoms;

The first two steps require a direct intervention by the user, the last one is performed by the software. Concord interface is divided in tabs corresponding to such steps.
3.1.1 The first phase: analysis of terms

In the first phase (fig. 1) the user reviews each term splitting it into its different parts corresponding to:

1. atom
2. stop word
3. impossible word

Fig. 1: The first tab – term analysis

A preliminary subdivision is proposed by the software and is based on the results of former analysis. In such a subdivision, the user can set:

- consider/ignore the part of a term closed into brackets
- case sensitive or case insensitive logic.

A full text search tool allows to locate any term in the term list by searching for the whole term or a part of it.

When an atom (e.g. “acid”) which is part of another one (e.g. acidic particle) has been already validated, the systems proposes a subdivision including a “+”. The result is like “acid+ic|particle”. If it is decided that “acidic” is an atom by itself, it is sufficient to manually delete the “+”. The same applies to any other automatic subdivision proposed by the software.

Once an atom is validated, it will be used to identify other similar atoms in other terms.

3.1.2 The second phase: analysis of atoms

The second phase allows the user to review the terms starting from the atoms in an atom-to-term interface. Reviewing atoms includes the definition of sub-atoms, i.e. prefixes and suffixes.
In the left window, the complete list of the atoms validated during the first phase is presented. In the right window all the elements that will be used for the validation of sub-atoms are presented. In the lower part of the window, the bar for the navigation and editing of the table of atoms is displayed. The navigation bar presents 10 buttons. Using this bar it will be possible to navigate through the table, insert or delete records, edit, confirm and undo an operation, update the table.

Once an atom is selected it is displayed in the window followed by a partial index made by the concordances of the selected atom and by the terms containing it.

It is now possible, using the Concord symbolism, to make changes to the atom structure and divide it into sub-atoms. A sub-atom could be a prefix or a suffix.

To identify a prefix, the left angle bracket “<” is used so that all the characters preceding the symbol are interpreted as belonging to a prefix sub-atom. Following this methodology the term “biome” becomes “bio<me” and the system considers “bio” as a prefix and “me” is not considered.

To identify a suffix, the right angle bracket “>” is used so that all the characters following the symbol are interpreted as belonging to a suffix sub-atom. Following this methodology the term “absorption” becomes “ab>sorption” and the system considers “sorption” as a suffix and “ab” is not considered.
There are more complex cases, like in the term “chlorobenzene” where two sub-atoms appear of which one is the prefix (“chloro”) and the other is a suffix (“benzene”). In this case a combination of the two symbols is used (“chloro<>benzene”). In this context the terms “prefix” and “suffix” are used referring to a simple string logic.

![Subdivision of “chlorobenzene”](image)

*Fig. 4: The result of the subdivision of “chlorobenzene” in sub-atoms*

When a term contains not only a prefix and a suffix but also a final part that has to be ignored like in “bioacidic”, the symbol “.” is used (bio<>acid.ic”). The symbols “<” and “>” could also be used in combination like in “agrifoodstuff”; the resulting string is “agri<>food<stuff” where “agri” is a prefix, “food” is a suffix of “agri” and a prefix of “stuff” while “stuff” is not considered.

### 3.1.3 The third phase: the index

Once the phase 2 is completed, next step is represented by the generation of the final index and to export the results in text format. It is sufficient to click on “File, export” choosing the name of the file to be generated.

### 4. Results, their use and future development

Concord can be easily applied to thesauri, dictionaries and any other lexical/terminological content since it refers to standard databases and allows parametric configuration. It is foreseen the application of Concord also to microthesauri on which EKOLab is currently working like the GIS and Remote Sensing microthesaurus and the SnowTerm project. Another point under development is the integration at the level of tables between Concord and other software like SuperThes. Once implemented, this will allow to work using the same database storing data and creating internal links.

### Bibliography
