UNL Lexical Selection with Conceptual Vectors

Mathieu LAFOURCADE*, Christian BOITET**

*LIRMM
161, rue Ada
F-34392 Montpellier cedex 5, France
Mathieu.Lafourcade@lirmm.fr

**GETA, CLIPS, IMAG
385, av. de la bibliothèque, BP 53
F-38041 Grenoble cedex 9, France
Christian.Boitet@imag.fr

Abstract

When deconverting a UNL graph into some natural language LG, we often encounter lexical items (called UWs) made of an English headword and formalized semantic restrictions, such as "look for (icl>do, agt>person)", which are not yet connected to lemmas, so that is it necessary to find a "nearest" UW in the UNL-LG dictionary, such as "look for (icl>action, agt>human, obj>thing)". Then, this UW may be connected to several lemmas of LG. In order to solve these problems of incompleteness and polysemy, we are applying a method based on the computation of "conceptual vectors", previously used successfully in the context of thematic indexing of French and English documents.

Keywords: disambiguation, deconversion, UNL-French localization, transfer, conceptual vectors, lexical selection

Introduction

The UNL project of network-oriented multilingual communication has proposed a standard for encoding the meaning of natural language utterances as semantic hypergraphs intended to be used as pivots in multilingual information and communication systems. In the first phase (1997-1999), more than 16 partners representing 14 languages have worked to build deconverters transforming an (interlingual) UNL hypergraph into a natural language utterance.

The UNL-French deconverter first performs a "localization" operation within the UNL format, and then classical transfer and generation steps (Boitet & al., 1982; Boitet, 1997; Slocum, 1984). This raises interesting issues about the status of the UNL language, designed as an interlingua, but diversely used as a linguistic pivot (disambiguated abstract English), or as a purely semantic pivot.

When deconverting a UNL graph into some natural language LG, we often encounter lexical items (called UWs) made of an English headword and formalized semantic restrictions, such as `"look for (icl>do, agt>person)"`, which are not yet connected to lemmas, so that is it necessary to find a "nearest" UW in the UNL-LG dictionary, such as `"look for (icl>action, agt>human, obj>thing)"`. Then, this UW may be connected to several lemmas of LG. In order to solve these problems of incompleteness and polysemy, we apply a method based on the computation of "conceptual vectors", previously used successfully in the context of thematic indexing of French and English documents.

We first present our general technique of disambiguation using conceptual vectors (DCV), then the context of disambiguation in a deconversion from UNL into a natural language, and the application of DCV to this problem.

1. Conceptual Vectors

1.1 Outline of the method

In short, our method is as follows. First, we prepare a very large dictionary of wordsenses with associated conceptual vectors. We begin by associating very "crude" conceptual vectors manually to a small set of terms, our "kernel". The dimensions are the 873 leaves of Roget's thesaurus for English, adapted to French. We can also "unfold" some of these dimensions into more detailed specific thesaurii.

We then use a large coverage French analyzer to transform all definitions of all the terms known by the analyzer into annotated tree structures. Then, we attach the crude conceptual vectors to the kernel terms, and empty conceptual vectors to all other words and all non lexical nodes, and perform simulated annealing on the whole tree. The conceptual vector of the root becomes the conceptual vector for the word sense in question, while the conceptual vectors of non kernel terms become new initial vectors for them. This way, the kernel grows.

In December 2001, we had 64,000 terms, an average of 3.3 word senses (definitions) per term, and 210,000 conceptual vectors.

We use several distances between conceptual vectors, among them the classical Arg_cosine, which has a natural interpretation in terms of "angular distance" and models well the notion of "distance from a point of view". This particular distance is used to classify the conceptual vectors of each term into a binary decision tree. The leaves contain the conceptual vectors of the individual definitions and the internal nodes a weighted average of the conceptual vectors of their daughters. This is useful because we use all kinds of dictionaries, with the result that two definitions may be different but very close.

This "learning process" is iterated constantly over the growing set of terms.

To disambiguate a particular occurrence of a term in a document, we first analyze the whole document into a possibly large decorated tree (several dozen pages are routinely processed as one tree). We then attach to the lexical nodes their average conceptual vectors, and perform simulated annealing on the document tree. The conceptual vectors near the top of the tree give a thematic characterization of the corresponding parts of the document (section, paragraph…).

The conceptual vector of each lexical node has also changed into a "contextually recooked" vector. It is now possible to find the closest conceptual vector in its binary decision tree. This "contextual CV-based disambiguation process" produces either a set of possible senses (the leaves of that subtree), or one sense (the closest among them).

1.2 Mathematical basis

1.2.1 Conceptual vector space

The conceptual vector model is based on the projection on a mathematical model of the linguistic notion of semantic fields. The question of how to choose (or build) a concept set is far beyond the scope of this model and is left to people studying ontologies. In our prototype applied to French and English, we have chosen (Larousse 1992) where 873 concepts are identified.

The main hypothesis is that this set constitutes a generator space for the words (terms in general) and their meanings and as such, any word would project its meanings on this space.

Let C be a finite set of n concepts. A conceptual vector V is a linear combination of elements of C. For a meaning A, vector V_A is the description (in extension) of activations of all concepts of C. For example, the different meanings of *to tidy up* and of *to cut* could respectively be projected on concepts of C as follows (for clarity sake, CONCEPT [intensity] are ordered by decreasing intensity values).

V(to tidy up) = CHANGE [0.84], VARIATION [0.83], EVOLUTION [0.82], ORDER [0.77], SITUATION [0.76], STRUCTURE [0.76], RANK [0.76] …

V(*to cut)* = GAME [0.8], LIQUID [0.8], CROSS [0.79], PART [0.78] MIXTURE [0.78], FRACTION [0.75], TORTURE [0.75] WOUND [0.75], DRINK [0.74] …

Lexical items associated with their vectors are stored in conceptual lexicons. Each meaning of a polysemous word is associated to a different vector. The global vector of a term is (with some simplification) a normalized vector sum of all its meanings. For instance:

V(head) = HEAD [0.83], . BEGINNING [0.75], ANTERIORITY [0.74], PERSON [0.74] INTELLIGENCE [0.68], HIERARCHY [0.65], …

The following metaphor may help apprehending why the angular distance can be used as an artifact for thematic proximity. Let us see the space of all word senses as a sky full of stars. The empty space between two stars may be pointed to although there is no star (word sense) between them. Stars form constellations, some parts of the space being crowded, others being underpopulated. Then, a meaning is a direction in the space, but not an actual point, as, from the observer point of view, the Euclidean distance between the observer and the point cannot be assessed. The angle between two directions defines their distance.

We don't consider the vector norm for the following reason. Take a vector representing the idea of the red color. Take another vector collinear but with twice its norm. Does the second vector represent the idea of something redder? If yes, then the first one is less red, which means that it might be more blue (or yellow or green or darker or lighter, etc.). But, in this case, it should not point to the same direction, which is not what we supposed at first. The vector norm may be used as a measure of the intensity of expression of the idea (like from screaming to whispering) but not directly as an estimator of thematic activation and closeness.

1.2.2 Distance and test functions

We define Sim(X, Y) as one of the similarity measures often used in information retrieval (Morin 99). Using this measure, we can express the angular distance D_A between two vectors X and Y by:

$$D_A(X,Y) = \operatorname{Arc\,cos}(\operatorname{Sim}(X,Y)) = \operatorname{Arc\,cos}\left(\frac{X \cdot Y}{\|X\| \times \|Y\|}\right)$$

Intuitively, this function constitutes an evaluation of the thematic proximity. Mathematically, it is the measure of the (hyper)angle between the two vectors. We consider, that, if $D_A(X, Y) \le \pi/4$, X and Y are thematically close and share many concepts. For $D_A(X, Y) \ge \pi/4$, the semantic proximity between X and Y is considered as loose. Around $\pi/2$, meanings are almost without any relation. At $\pi/2$, they have strictly no relationship (which never happens in practice).

This is a real distance function (contrary to the similarity measure) as it verifies the properties of reflexivity, symmetry and triangular inequality.

$$\begin{cases} D_A(X, X) = 0 \\ D_A(X, Y) = D_A(Y, X) \\ D_A(X, Y) + D_A(Y, Z) \ge D_A(X, Z) \end{cases}$$

We have by definition $D_A(0, 0) = 0$ and $D_A(X, 0) = \pi/2$ with 0 as the null vector. The null vector has no associated word in any language, as it represents the "empty idea", which does not activate any concept.

Let X be a lexical property. We define the test function $P_x(V)$ of V against X as:

$$P_p(X) = \frac{\pi}{2} - D_A(V(p), X)$$

We use test functions to give a score to lexical items in inverse proportion of their distance to (the set of words meeting some) lexical constraints. In the context of UNL, these properties will be the UNL restrictions as expressed in the UWs.

1.2.3 Useful vector operations

The normalized vector sum of X and Y is the vector V defined by:

$$V = X \oplus Y \quad \Big| \quad v_i = \frac{x_i + y_i}{\|V\|}$$

The sum can be generalized to any number of vectors:

$$V = \sum_i X_i \quad \Big| \quad v_i = \frac{\sum x_i}{\|V\|}$$

The term to term vector product of X and Y is the vector V defined by:

$$V = X \otimes Y \quad \Big| \quad v_i = \sqrt{x_i y_i}$$

We can interpret the sum as the mean (or barycenter) of the vectors. The normalized term to term product can be seen as a kind of intersection between vector components. Note that the norm of the resulting vector of the product is lower or equal to 1.

1.3 Lexical contextualization

Outside of any context, when a word *w* has *n* meanings, it is associated to *n* vectors Vi and the global vector of *w* is the barycenter of all Vi (with weights all set to 1). The construction of a contextualized vector V is done by modifying these weights according to the context. It is then a vector sum where weights are Pp(X) values :

$$Vp(w) = \sum_{i=0}^{n} P_p(V_i(w)).V_i(w)$$

For instance, the vector associated to the (highly) polysemic word *head* in the context of P_{body} refers properly to the body part.

$V_{body}(head)$ = HEAD [0.97], . PERSON [0.85] INTELLIGENCE [0.78], BODY [0.75], …

2. UNL-French deconversion

2.1 The UNL project and language

2.1.1 The project

The pivot paradigm is used: the representation of an utterance in the UNL interlingua (UNL stands for "Universal Networking Language") is a hypergraph where normal nodes bear UWs ("Universal Words", or interlingual acceptions) with semantic attributes, and arcs bear semantic relations (deep cases, such as agt, obj, goal, etc.). Hypernodes group a subgraph defined by a set of connected arcs. A UW denotes a set of interlingual acceptions (word senses), although we often loosely speak of "the" word sense denoted by a UW.

Because English is known by all UNL developers, the syntax of a normal UW is: "<English word or compound> (<list of restrictions>)", e.g. "look for (icl>action, agt>human, obj>thing)".

Going from a text to the corresponding "UNL text" or interactively constructing a UNL text is called "enconversion", while producing a text from a sequence of UNL graphs is called "deconversion".

This departure from the standard terms of analysis and generation is used to stress that this is not a classical MT project, but that UNL is planned to be the source format preferred for representing textual information in the envisaged multilingual network environment. The schedule of the project, beginning with deconversion rather than enconversion, also reflects that difference.

Each group is free to use its own software tools and/or lingware resources, or to develop directly with tools provided by the UNL Center (UNU/IAS & UNDL).

Emphasis is on a very large lexical coverage, so that all groups spend most of their time on the UNL-NL lexicons, and develop tools and methods for efficient lexical development. By contrast, grammars have been initially limited to those necessary for deconversion, and are gradually expanded to allow for more naturalness in formulating text to be enconverted.

2.1.2 The UNL components

2.1.2.1 Universal Words

The nodes of a UNL utterance are called Universal Words (or UWs). The syntax of a normal UW consists of a headword and a list of restrictions.

Because English is known by all UNL developers, the headword is an English word or compound. The restrictions are given as an attribute value pair where attributes are semantic relation labels (those used in the graphs and some more thesaurus-oriented) and values are other UWs (restricted or not).

A UW denotes a collection of interlingual acceptions (word senses), although we often loosely speak of "the" word sense denoted by an UW. For example, the unrestricted UW “`look for`” denotes all the word-senses associated to the English compound word “look for”. The restricted UW `"look for(icl>action, agt>human, obj>thing)"` represents all the word senses of the English word “look for” that are an action, performed by a human that affects a thing. In this case this leads to the word sense: “look for – to try to find”.

2.1.2.2 UNL hypergraphs

A UNL expression is a hypergraph with a unique entry node (a connected graph may be labelled and given an entry node, thereby becoming a subgraph or "scope"). The arcs bear semantic relation labels (deep cases, such as agt, obj, goal, etc.).

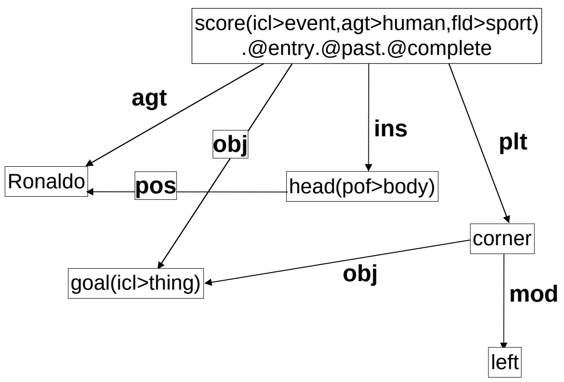

Fig. 1: a possible UNL graph for “Ronaldo has headed the ball into the left corner of the goal”

In a UNL graph, UWs appear with attributes describing what is said from the speaker’s point of view. This includes phenomena like speech acts, truth values, time, etc.

These hypergraphs are written as text using the UNL "language": a graph is written as a list of arcs, connecting the different nodes. For example, the graph presented in Fig. 1 can be written as:

```
agt(score(…).@entry.@past.@complete,
    Ronaldo)
obj(score(…).@entry.@past.@complete,
    goal(icl>thing))
pof(head(pof>body),
    Ronaldo)
ins(score(…).@entry.@past.@complete,
    head(pof>body))
plt(score(…).@entry.@past.@complete,
    corner)
obj(corner, goal(icl>thing))
mod(corner, left)
```

Fig. 2: Linear writing of the same UNL graph

2.2 The place of disambiguation in the French deconverter

2.2.1 Overview

The global deconversion process for French (without conceptual vectors) has been presented in (Sérasset & Boitet, 1999). Deconverting consists in transforming a UNL graph into one or more utterances in a natural language. To this purpose, we segment the process into 7 phases, as illustrated below. The third phase (graph-to-tree) produces a decorated tree which is fed into an Ariane-G5 TS (structural transfer).

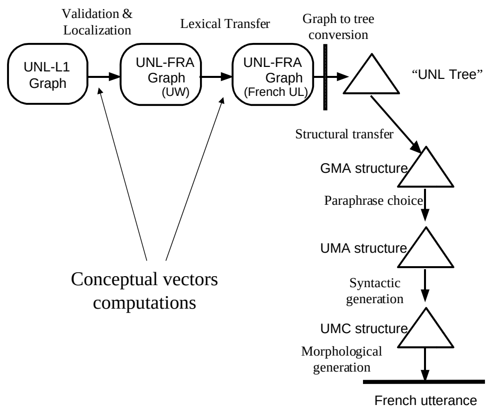

Fig. 3: architecture of the French deconverter with lexical selection enhanced using conceptual vectors

2.2.2 Transfer

2.2.2.1 Validation

When a UNL graph is to be deconverted, its correctness is first checked: it has to be connected, and the features on the nodes must be in a predefined list. Validation is necessary to improve the robustness of the deconverter, as there is no hypothesis on the way a graph is created. Invalid graphs are rejected.

2.2.2.2 Lexical & cultural localization

Some lexical units used in the graph may not be present in the French deconversion dictionary. This problem may appear under different circumstances. First, the French dictionary (constantly under development) may be incomplete. Second, the UW may use an unknown notation to represent a known French word sense. Third, the UW may represent a word sense absent from the French language.

We solve these problems with the following method : Let w be a UW in the graph G. Let D be the French dictionary (a set of UWs). We substitute w in G by w’ such that : $w' \in D$ and $\forall x \in D\ d(w, w', G) = d(w, x, G)$. where d is a pseudo-distance function.

Without vectors, if different French UWs are at the same pseudo-distance of w, w’ is chosen at random among these UWs (default in non-interactive mode).

On the “cultural” aspect, some crucial information may be missing, depending on the language of the source utterance (sex, modality, number, determination, politeness, kinship, etc).

It is in general impossible to solve this problem fully automatically in a perfect manner, as we do not know anything about the document, its context, and its intended usage: FAHQDC[1] is no more possible than FAHQMT on arbitrary texts. We have to rely on necessarily imperfect heuristics. The heuristics we have chosen use conceptual vector contextualization.

2.2.2.3 Lexical transfer

After the localization phase, we perform a lexical transfer. It would seem natural to convert the graph into a tree and then to do it in Ariane-G5. But lexical transfer is context-sensitive, and we want to avoid the possibility of transferring differently two tree nodes corresponding to one and the same graph node.

Each graph node is replaced by a French lexical unit (LU), along with some variables. A lexical unit used in the French dictionary denotes a derivational family (e.g. in English: destroy denotes destroy, destruction, destructible, destructive…, in French: détruire covers détruire, destruction, destructible, indestructible, destructif, destructeur). This is where conceptual vectors can help to select the most probable meaning according to the (vector) context.

There may be several possible lexical units for one UW. This happens when there is a real synonymy or when different terms are used in different domains to denote the same word sense[2]. In that case, without conceptual vectors the choice of the lexical unit is done at random or interactively as there is no information about the task the deconverter is used for.

The same problem also appears because of the strategy used to build the French dictionary. In order to obtain a good coverage from the beginning, we have underspecified the UWs and linked them to different lexical units. This way, we considered a UW as the denotation of a set of word senses in French.

Hence, we can reuse previous dictionaries, and use the dictionary even if it is still under development and incomplete. In our first version, we also solve this problem by a random selection of a lexical unit.

2.2.2.4 Graph to tree conversion

The subsequent deconversion phases are performed in Ariane-G5. Hence, it is necessary to convert the UNL hypergraph into an Ariane-G5 decorated tree.

The UNL graph is directed. Each arc is labelled by a semantic relation (agt, obj, ben, con…) and each node is decorated by a UW and a set of features. One node is distinguished as the "entry" of the graph.

An Ariane-G5 tree is a general (non binary) tree with decorations on its nodes. Each decoration is a set of variable-value pairs.

The graph-to-tree conversion algorithm has to maintain the direction and labelling of the graph along with the decorations of the nodes.

Our algorithm splits the nodes that are the target of more than one arc, and reverses the direction of as few arcs as possible. An example of such a conversion is shown in figure 2.3.

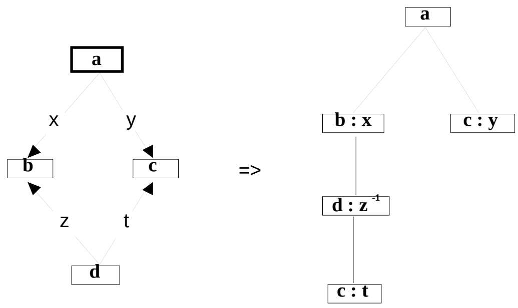

Fig. 4: example graph to tree conversion

The graph to tree conversion algorithm has been extensively described in (Sérasset & Boitet, 2000).

2.2.2.5 Structural transfer

The purpose of the structural transfer is to transform the tree obtained so far into a Generating Multilevel Abstract (GMA) structure (Boitet, 1994). In this structure, non-interlingual linguistic levels (syntactic functions, syntagmatic categories…) are underspecified, and (if present), are used only as a set of hints for the generation stage.

3. Integration of DCV in the deconversion process

3.1 DCV in localization

The vector contextualization generalizes both kinds of localization (lexical and cultural). The selected UW is the one which vector is the closest to the contextualized vector.

All restrictions of the UWs in the UNL graph are taken as information for the lexical contextualization. For example, the vector attached to the word *head* will be contextualized with *body* because of the presence of the restriction `pof>body`. Although different weights could be associated to the kind of restriction (as `icl`, `agt`, `fld`, …) we simply consider these restrictions equally, because we

[1] fully automatic high quality deconversion.

[2] strictly speaking, the same collection of interlingual word senses (acceptions).

cannot know in advance which are the most relevant.

Some restrictions are not valued (like `plt`) and are converted (from a correspondence table) into appropriate contexts (like *place* for `plt`).

3.2 Computing conceptual vectors on the tree

When the graph has been converted into a tree, a full conceptual vector analysis can be undertaken. Vectors are attached to the tree and then propagated up to the root. Usually, we back propagate toward the leaves, to induce a vector mutual activation. We usually perform cycles until the root vector converges. As convergence is not guaranteed in general, we set a maximum number of cycles to force the process to end.

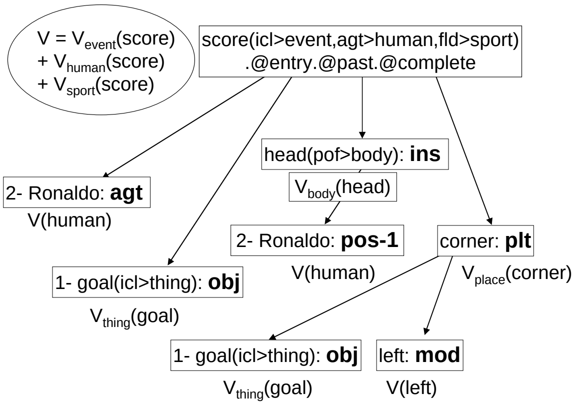

Fig. 5: Vector attachment to a UNL tree

In the upward process, the vector of a node N with k daughters n_1 … n_k is the sum of its original vector plus the mean (normalized sum) of all daughter vectors:

$$\uparrow V'(N) = V(N) \oplus \sum_{i=0}^{n} V(n_i)$$

For example, the vector of the node *corner* will be itself plus the sum of V(goal) (contextualized as an object) and V(left).

In the downward process, the vector is weakly contextualized by its father, by the application of the term to term product.

$$\downarrow V'(n_i) = V(n_i) \oplus V(N) \otimes V(n_i)$$

The global effect of this process is to make each vector “resonate” with both other vectors and the restriction, to produce a vector that is as close as possible to the different possible meanings and to the context.

3.3 DCV in lexical transfer

The transfer is a generalization of the lexical selection to another conceptual lexicon. From a (contextualized) vector V of a conceptual lexicon A, we select the item of the closest vector in the conceptual lexicon B. This strategy is used each time several lexical unit compete for one UW. We should note here that the same concept set has been used for the construction of the conceptual lexicon of English and French. This approach makes the comparison of vector feasible, although it may be questioned from an ontological point of view between two different languages.

Conclusion

When deconverting a UNL graph into a natural language LG, we often encounter lexical items (called UWs) made of an English headword and formalized semantic restrictions, which are not yet connected to lemmas, so that is it necessary to find a "nearest" UW in the UNL-LG dictionary. Then, this UW may be connected to several lemmas of LG.

In order to solve these problems of incompleteness and polysemy, we apply a method based on the computation of "conceptual vectors", previously used successfully in the context of thematic indexing of French and English documents.

References

Blanc É. & Guillaume P. (1997) *Developing MT lingware through Internet : ARIANE and the CASH interface*. Proc. of Pacific Association for Computational Linguistics 1997 Conference (PACLING'97), Ohme, Japon, 2-5 September 1997, 1/1, pp. 15-22.

Blanchon H. (1994) *Perspectives of DBMT for monolingual authors on the basis of LIDIA-1, an implemented mockup*. Proc. of 15th International Conference on Computational Linguistics, COLING-94, 5-9 Aug. 1994, 1/2, pp. 115—119.

Boitet C., Réd. (1982) *"DSE-1"— Le point sur ARIANE-78 début 1982*. Contrat ADI/CAP-Sogeti/Champollion (3 vol.), GETA, Grenoble, janvier 1982, 616 p. (200 p. + annexes)

Boitet C., Guillaume P. & Quézel-Ambrunaz M. (1982) *ARIANE-78, an integrated environment for automated translation and human revision*. Proc. of COLING-82, Prague, July 1982, North-Holland, Ling. series 47, pp. 19—27.

Boitet C. (1994) *Dialogue-Based MT and self-explaining documents as an alternative to MAHT and MT of controlled languages*. Proc. of Machine Translation 10 Years On, 11-14 Nov. 1994, Cranfield University Press, pp. 22.21—29.

Boitet C. & Blanchon H. (1994) *Multilingual Dialogue-Based MT for Monolingual Authors: the LIDIA Project and a First Mockup*. Machine Translation, Vol. 9, N° 2, pp. 99—132.

Boitet C. (1997) *GETA's MT methodology and its current development towards personal networking communication and speech translation in the context of the UNL and C-STAR projects*. Proc. of PACLING-97, Ohme, 2-5 September 1997, Meisei University, pp. 23-57. (invited communication)

Brown R. D. (1989) *Augmentation*. (Machine Translation), Vol., N° 4, pp. 1299-1347.

Ducrot J.-M. (1982) *TITUS IV*. In *Information research in Europe. Proc. of the EURIM 5 conf. (Versailles)*, edited by Taylor P. J., London, ASLIB.

Kay M. (1973) *The MIND system*. In *Courant Computer Science Symposium 8: Natural Language Processing*, edited by Rustin R., New York, Algorithmics Press, Inc., pp. 155-188.

Lafourcade M. (2001) *Lexical sorting and lexical transfer by conceptual vectors*. Proc. of MMA'01, 29-31/1/01, SigMatics & NII, Tokyo, 10 p.

Lafourcade M. & Prince V. (2001) *Synonymy and conceptual vectors*. Proc. of NLPRS-2001, Tokyo, 27-3011/01, pp. 127-134.

Larousse (1999) *Thésaurus Larousse - des idées aux mots - des mots aux idées*. In *In extenso*. Larousse, Paris. Edited by Péchouin D., 1146 p. (2ième édition)

Maruyama H., Watanabe H. & Ogino S. (1990) *An Interactive Japanese Parser for Machine Translation*. Proc. of COLING-90, 20-25 août 1990, ACL, 2/3, pp. 257-262.

Melby A. K., Smith M. R. & Peterson J. (1980) *ITS : An Interactive Translation System*. Proc. of COLING-80, Tokyo, 30/9-4/10/80, pp. 424—429.

Moneimne W. (1989) *TAO vers l'arabe. Spécification d'une génération standard de l'arabe. Réalisation d'un prototype anglais-arabe à partir d'un analyseur existant*. Nouvelle thèse, UJF. (159 p. + annexes)

Nirenburg S. & al. (1989) *KBMT-89 Project Report*., Center for Machine Translation, Carnegie Mellon University, Pittsburg, April 1989.

Nyberg E. H. & Mitamura T. (1992) *The KANT system: Fast, Accurate, High-Quality Translation in Practical Domains*. Proc. of COLING-92, 23-28 July 92, ACL, 3/4, pp. 1069—1073.

Sérasset G. & Boitet C. (2000) *On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter*. Proc. of COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL, 7 p. (submitted)

Slocum J. (1984) *METAL: the LRC Machine Translation system*. In *Machine Translation today: the state of the art (Proc. third Lugano Tutorial, 2–7 April 1984)*, edited by King M., Edinburgh University Press (1987).

Wehrli E. (1992) *The IPS System*. Proc. of COLING-92, 23-28 July 1992, 3/4, pp. 870-874.