Coedition to share text revision across languages and improve MT a posteriori

Christian BOITET
GETA, CLIPS, IMAG
385 rue de la Bibliothèque, BP 53
38041 Grenoble cedex 9, France
Christian.Boitet@imag.fr

TSAI Wang-Ju
GETA, CLIPS, IMAG
385 rue de la Bibliothèque, BP 53
38041 Grenoble cedex 9, France
Wang-Ju.Tsai@imag.fr

Abstract

Coedition of a natural language text and its representation in some interlingual form seems the best and simplest way to share text revision across languages. For various reasons, UNL graphs are the best candidates in this context. We are developing a prototype where, in the simplest sharing scenario, naive users interact directly with the text in their language (L0), and indirectly with the associated graph. The modified graph is then sent to the UNL-L0 deconverter and the result shown. If is is satisfactory, the errors were probably due to the graph, not to the deconverter, and the graph is sent to deconverters in other languages. Versions in some other languages known by the user may be displayed, so that improvement sharing is visible and encouraging. As new versions are added with appropriate tags and attributes in the original multilingual document, nothing is ever lost, and cooperative working on a document is rendered feasible. On the internal side, liaisons are established between elements of the text and the graph by using broadly available resources such as a L0-English or better a L0-UNL dictionary, a morphosyntactic parser of L0, and a canonical graph2tree transformation. Establishing a "best" correspondence between the "UNL-tree+L0" and the "MS-L0 structure", a lattice, may be done using the dictionary and trying to align the tree and the selected trajectory with as few crossing liaisons as possible. A central goal of this research is to merge approaches from pivot MT, interactive MT, and multilingual text authoring.

Keywords: revision sharing, interlingual representation, text / UNL coedition, multilingual communication

Résumé

La coédition d'un texte en langue naturelle et de sa représentation dans une forme interlingue semble le moyen le meilleur et le plus simple de partager la révision du texte vers plusieurs langues. Pour diverses raisons, les graphes UNL sont les meilleurs candidats dans ce contexte. Nous développons un prototype où, dans le scénario avec partage le plus simple, des utilisateurs "naïfs" interagissent directement avec le texte dans leur langue (L0), et indirectement avec le graphe associé. Le graphe modifié est ensuite envoyé au déconvertisseur UNL-L0 et le résultat est affiché. S'il est satisfaisant, les erreurs étaient probablement dues au graphe et non au déconvertisseur, et le graphe est envoyé aux déconvertisseurs vers d'autres langues. Les versions dans certaines autres langues connues de l'utilisateur peuvent être affichées, de sorte que le partage de l'amélioration soit visible et encourageant. Comme les nouvelles versions sont ajoutées dans le document multilingue original avec des balises et des attributs appropriés, rien n'est jamais perdu, et le travail coopératif sur un même document est rendu possible. Du côté interne, des liaisons sont établies entre des éléments du texte et du graphe en utilisant des ressources largement disponibles comme un dictionnaire L0-anglais, ou mieux L0-UNL, un analyseur morphosyntaxique de L0, et une transformation canonique de graphe UNL à arbre. On peut établir une "meilleure" correspondance entre "l'arbre-UNL+L0" et la "structure MS-L0", une treille, en utilisant le dictionnaire et en cherchant à aligner l'arbre et une trajectoire avec aussi peu que possible de croisements de liaisons. Un but central de cette recherche est de fusionner les approches de la TA par pivot, de la TA interactive, et de la génération multilingue de texte.

Mots-clés: révision partagée, représentation interlingue, coédition texte / UNL, communication multilingue

Introduction

Creating and maintaining aligned multilingual documents is a growing necessity. In the current practice, a multilingual document consists in many parallel monolingual files, which may be technical documentation as well as help files, message files, or simply thematic information put on the web and intended for a multilingual audience (medicine, cooking, travel…). The task is difficult even for a document managed in a centralized manner. Ususally, it is first created in a unique source language, and translated into several target languages. There must be a way to keep trak of modifications, possibly done at various places on different linguistic versions. From time to time, somebody has to decide which modifications to integrate in the next release of the document. For that, modifications done in target languages have to be translated back into the source language. The new and the old source versions are then compared using (fuzzy) matching techniques, so that only really new segments are sent for translation.

The problem arises even more if the documents are not managed centrally, so that the monolingual files are often in various formats (Word, EgWord, Interleaf, FileMaker, DBMS formats, etc.).

A. Assimi [1, 2] has shown how to "realign" parallel decentralized documents and apply the methodology sketched above. However, in both cases, human translators have to retranslate the modified or new source segments, or to revise them if they are retranslated by a quality MT system. Contrary to what is often said, quality MT exists, but for specific contexts only. (See [14]).

What we would like to do is to make it possible to share the revision work across languages, whatever the domain and the context. It is clearly impossible to reflect changes on a file in language L0 into files in L1,… Ln automatically and faithfully, without any intermediate structure to bridge the gap, because that would necessitate at least a perfect fine-grained aligner in case of changing articles or common nouns (provided the gender and number stay the ame in each Li version). In case of replacing a verb by another with a different valency frame in a target Li, the sentence in Li would have to be reanalyzed, transformed accordingly, and regenerated without introducing any new error or imprecision, thereby keeping the manual improvements coming from previous manual revisions. Or we would need a more than perfect MT system, namely one which would be able to analyze the changed utterance in L0, and to transfer and generate it into a sentence of Li as close as possible as the previous sentence in Li, which again could have been improved manually before.

The best and simplest way to go seems to use some formalized interlingua IL and to

(1) reflect the modifications from L0 to the IL,

(2) regenerate into L1,… Ln from the IL.

We should also allow for direct manual improvements, considering that the IL form will not always be present, or not always improvable enough for lack of expressivity, or that generators will never be perfect. We choose UNL [3, 4, 10, 11] as our IL of choice for various reasons:

(1) it is specifically designed for linguistic and semantic machine processing,

(2) it derives with many improvements from H.Uchida's pivot used in ATLAS-II (Fujitsu) [13], still evaluated as the best quality MT system for English-Japanese, with a large coverage (586,000 lexical entries in each language),

(3) participants of the UNL project[1] have built "deconverters" from UNL into about 12 languages, and at least the Arabic, Indonesian, Italian, French, Russian, Spanish, and Thai deconverters were accessible for experimentation through a web interface at the time of writing,

(4) although formal, UNL graphs (see below) are quite easy to understand with little training and may be presented in a "localized" way to naive users by translating UNL symbols (semantic relations, attributes) and lexemes (UWs) into symbols and lexemes of their language,

(5) the UNL project has defined a format embedded in html for files containing a complete multilingual document aligned at the level of utterances, and produced a "visualizer" transforming a UNL file into as many html files as languages, and sending them to any web browser.

The UNL representation of a text is a list of "semantic graphs", each expressing the meaning of a natural language utterance. Nodes contain lexical units and attributes, arcs bear semantic relations. Connex subgraphs may be defined as "scopes", so that a UNL graph may be a hypergraph.

The lexical units, called Universal Words (UW), represent (sets of) word meanings, something less ambitious than concepts. Their denotations are built to be intuitively understood by developers knowing English, that is, by all developers in NLP. AUW is an English term or special symbol (number…) possibly completed by semantic restrictions : the UW "process" represents all word meanings of that lemma, seen as citation form (verb or noun here), and "process(icl>do, agt>person)" covers only the meanings of processing, working on, etc.

The attributes are the (semantic) number, genre, time, aspect, modality, etc., and the 40 or so semantic relations are traditional "deep cases" such as agent, (deep) object, location, goal, time, etc.

One way of looking at a UNL graph corresponding to an utterance in language L is to say that it represents the abstract structure of an equivalent English utterance "seen from L", that is, where semantic attributes not necessarily expressed in L may be absent (e.g., aspect coming from French, determination or number from Japanese, etc.).

We will first present scenarios of increasing internal complexity for the situation where somebody reads a UNL document in her language, corrects it, and wants the corrections to carry over to the corresponding fragment in other languages. We will then study more precisely the correspondence between a text in language L0 and its representation in UNL, and show the advantage of breaking it into 3 parts: text ↔ morpho-syntactic lattice or chart ↔ abstract "UNL-tree" ↔ UNL graph. Finally, we present the current status of this work: an experimentation web site, a method to establish the second part of the correspondence, and related research.

[1] http://unl.ias.unu.edu

1. Scenarios for sharing revision across languages

Suppose a collection of multilingual documents is stored on a server as multilingual files in UNL-html format, or in any other form, e.g. in a data base, provided (1) it is possible to easily produce the version in any language contained in the document, (2) the versions are aligned at the level of utterance-like segments (a segment may contain more than 1 utterance), (3) UNL-graphs may be stored and aligned with the segments. Here is a slightly simplified example of a file in UNL-html format.

```
<HTML><HEAD><TITLE>Example 1  El/UNL
</TITLE></HEAD><BODY>
[D:dn=Mar Example 1, on= UNL French, mid=First.Author@here.com]
[P][S:1]{org:el}I ran in the park yesterday.{/org}
{unl}agt(run(icl>do).@entry.@past,i(icl>person))
plc(run(icl>do).@entry.@past,park(icl>place).@def)
tim(run(icl>do).@entry.@past,yesterday){/unl}
{cn dtime=20020130-2030, deco=man}
我昨天在公園裡跑步                                {/cn}
{de dtime=20020130-2035, deco=man}
Ich lief gestern im Park. {/de}
{es dtime=20020130-2031, deco=UNL-SP}
Yo corri ayer en el parque.{/es}
{fr dtime=20020131-0805, deco=UNL-FR}
J'ai couru dans le parc hier. {/fr}[/S]
[S:2]{org:el}My dog barked at me.{/org}{unl}
agt(bark(icl>do).@entry.@past,dog(icl>animal))
gol(bark(icl>do).@entry.@past,i(icl>person))
pos(dog(icl>animal),i(icl>person))
{/unl}{de dtime=20020130-2036, deco=man}
Mein Hund bellte zu mir.{/de}
{fr dtime=20020131-0806, deco=UNL-FR}
Mon chien aboya pour moi. [/S] [/P][/D]
</BODY></HTML>
```

Italian, Russian, French and Hindi. Hindi and Russian are not shown, but Japanese has been added by hand. The XML form is simplified.

Correct sentences are produced by the deconverters from correct and complete UNL graphs. We suppose here that the UNL graph has been produced from a Chinese version, and does not countain definiteness and aspectual information. Now all results are wrong wrt articles, and some wrt aspect.

```
<unl:S num="1">
<unl:org lg="cn">在博覽會之後，城市 將獲得一片海岸域 </unl:org>
<unl:unl>
<unl:arc> agt(retrieve(icl>do).@entry.@future, city) </unl:arc>
<unl:arc> tim(retrieve(icl>do).@entry.@future, after) </unl:arc>
<unl:arc> obj(after, Forum) </unl:arc>
<unl:arc> obj(retrieve(icl>do).@entry.@future, zone(icl>place).@indef) </unl:arc>
<unl:arc> mod(zone(icl>place).@indef, coastal) </unl:arc> </unl:unl>
<unl:cn> 在博覽會之後，城市 將獲得一片海岸域 </unl:cn>
<unl:el> After a Forum, a city will retrieve a coastal zone.</unl:el>
<unl:es> Ciudad recobrará una zona de costal después Foro. </unl:es>
<unl:fr> Une cité retrouvera une zone côtière après un forum. </unl:fr>
<unl:it> Città ricuperarà une zona costiera dopo Forum. </unl:it>
<unl:jp> フォーラムの後で, 都市は沿岸水域を取り出す。  </unl:jp>
</unl:S>
```

The French versions have been produced automatically, the German and Chinese manually. The output of the UNL viewer for French is:

```
<HTML><HEAD><TITLE>
Example 1  El/UNL
</TITLE></HEAD><BODY>
J'ai couru dans le parc hier.
Mon chien aboya pour moi.
</BODY></HTML>
```

and will probably be displayed by a browser as:

Example 1 El/UNL

J'ai couru dans le parc hier. Mon chien aboya pour moi.

and similarly for all other languages. In all scenarios, the user is reading the text in the normal display, not seing any tags, and wants to make some modification, such as moving "hier" after "couru" and changing "pour" to "vers". Activating some button or menu item, she enters a revision interface.

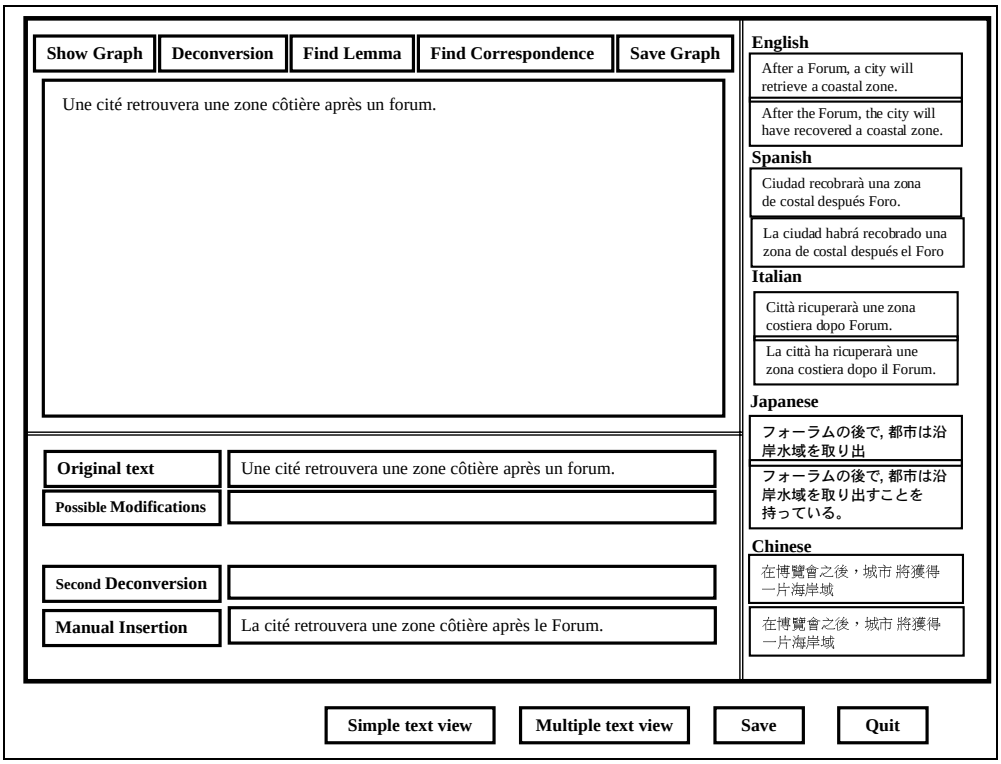
1.1 Multiple revision without sharing

In this first scenario, we don't suppose that there are UNL graphs associated with the segments. The problem is to transmit and add the user's modifications to the original form of the multilingual document. That is impossible by editing the html documents displayed, because they have no links to the original form. The UNL-html format predates XML, hence the special tags like [S] and {unl}, but we may transform it into an equivalent "UNL-xml" format. Then, using DOM and javaScript, it is possible to produce various views: that of a viewer, a bilingual or multilingual editable presentation, and a revision (coedition) interface.

This is an example from an experiment performed for the "Forum Barcelona 2004" on Spanish,

The following interface, designed to be used with sharing, may also be used by a reader knowing several languages, displayed on demand.

For example, a native Spanish speaker knowing French and English would put the correct articles ("La ciudad", "La cité", "The city", etc.) and the perfective aspect ("habra recobrado", "will have recovered"), but a native French speaker would probably not correct the aspect in English and Spanish, because aspect is often underspecified in French, e.g. in "retrouvera".



1.2 Transparent revision with sharing

In the second scenario, there is a UNL graph associated with the modified segment. In order to share the revisions across languages, we should reflect them on the UNL graph, e.g.

- add ".@def" on the nodes "city" & "Forum".
- replace "retrieve" by "recover" and add ".@complete" on the node containing it.

It is not possible in principle to deduce the modification on the graph from a modification on the text. For example, replacing "un" ("a") by "le" ("the") does not entail that the following noun is determined (.@def), because it can also be generic ("il aime la montagne" = "he likes mountains"). Hence, the technique envisaged is that:

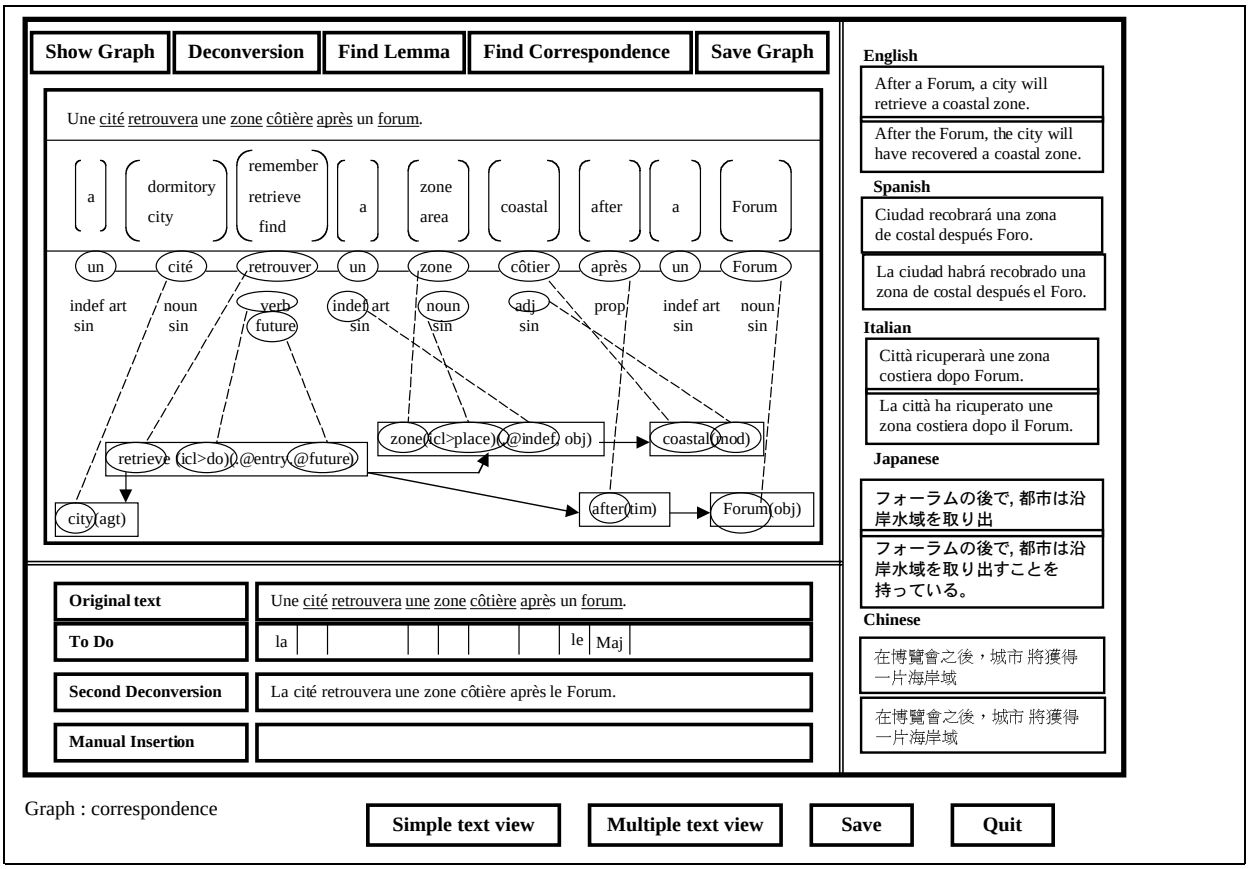
- revision is not done by modifying directly the text, but by using a menu system,
- the menu items have a "language side" and a hidden "UNL side",
- when a menu item is chosen, only the graph is transformed, and the action to be done on the text is stored and shown next to its focus.
- at any time, the new graph may be sent to the L0 deconverter and the result shown. If is is satisfactory, that shows that errors were due to the graph and not to the deconverter, and the graph may be sent to deconverters in other languages. Versions in some other languages known by the user may be displayed, so that improvement sharing is visible and encouraging.

New versions will be added with appropriate tags and attributes in the multilingual document in UNL-xml format, or in a DBMS, so that nothing is lost, and cooperative working on a document is feasible.

1.3 Revision on more than the texts

For the above method to work, the text has to be preprocessed, at least by computing morpho-syntactic classes (POS & actualization attributes) to avoid many spurious menus, segmenting, and lemmatizing. Because we want our technique to be widely applicable, this preprocessing should be such that it can be performed by large coverage tools freely available for many languages. That is the case for morphosyntactic analyzers (MSA), but not yet for full or even shallow parsers.

We also propose that the revision interface should allow access not only to the texts, but to editable representations of the UNL graph, of the result of the MSA, and of any other available structure such as a tree derived from the UNL graph.



For users not wanting to see anything else than text, the previous scenario will always be usable. But there are good reasons to "open the black box":

(1) the UNL Spanish group has successfully experimented with an interface for interactive UNL graph creation using a MSA and a graph editor showing the UNL graph in a "localized" way (symbols & lexemes appear in Spanish),

(2) it is sometimes much quicker to change something on another representation than on a text: for example, to merge two nodes in order to change "Mary likes Mary's daughter" into "Mary likes her daughter",

(3) it may even be necessary, if the correspondence is faulty and can not be improved because the text is very far from any reasonable deconversion obtainable from the graph,

(4) user interface technology has made much progress, and offers tools to build user-friendly direct manipulation environments,

(5) last but not least, the younger generation manipulates complex interfaces very naturally and expertly, far better than its elders!

1.4 What can and cannot be done

We identify 4 common types of errors in the corpus we have analysed so far:

(1) graphs containing false information: wrong attachment, wrong choice of UW, wrong attribute, wrong semantic relation…

(2) graphs with missing information, as above,

(3) absence of text because the UNL graph is formally incorrect (due to some wrong human manipulation, some bug in a deconverter…): missing parenthesis, missing entry node in a scope, disconnected graph…,

(4) deconversion errors.

Our method can be used for correcting the first 2 types of errors only. If a graph is formally incorrect, it may displayable or not. In the first case, it should be possible to manipulate and correct it graphically, e.g. by connecting 2 disconnected parts or choosing an entry node. In the second case, it is necessary to work on a textual representation. If errors come from the deconverter, the user may still correct the text by hand (last zone).

2. Establishing a text↔graph correspondence

2.1 The nature of correspondences

The correspondence between a text and a UNL graph may be decomposed into less complex liaisons, which are often not simple links, even between words and nodes. We found the following types in this case.

MS level	UNL graph
lemma arbre (French)	UW headword "tree"
lemma жениться (Russian)	complete UW marry(agt>male)
morpheme -tion (French, English) "男 " (Chinese "nan2")	restriction (icl>action) (agt>male)
particle "了 "(Chinese)	attribute .@complete
MS actualization feature plural	attribute .@pl
MS semantic feature his	relation pos(*, he)

2.2 Division in 3 subcorrespondences

We have already begun to break down the correspondence in 2 parts: text ↔ MS-structure ↔ UNL graph. The MS structure may always be embedded in a loop-free graph with information on the nodes (lattice) or on the arcs (charts), so that the first part of the correspondence is made of liaisons between substrings of the text (not necessarily always connex) and elements (nodes or arcs) on the trajectory corresponding to the preferred interpretation (in case of ambiguity).

It is perhaps possible to compute a direct correspondence between the MS lattice and the UNL graph, but it is not clear how to represent the liaisons between phrases and subgraphs. For that purpose, a tree structure is far better. Because there is no available large-scale and free syntactico-semantic analyzer for the vast majority of languages, we can not use even a tree produced by a shallow parser. But it is possible to associate a "standard UNL-tree" to any UNL graph by a reversible algorithmic transformation [3, 4, 10]: start at the outer entry node, and traverse the graph and its scopes (subgraphs) recursively, thereby creating auxiliary nodes for scopes, "inverse" semantic relations for arcs in the "wrong" direction, and coindexing symbols to represent reentrancy without duplication.

We can also take advantage of having one more structure by enriching it with lexical units of L0. Now the correspondence is broken into 3 parts:

- text ↔ MS-L0 (a lattice or a chart),
- MS-L0 ↔ UNL-tree+L0 (an unordered abstract quasi-dependency tree), and
- UNL-tree+L0 ↔ UNL-graph (liaisons may be produced by modifying the standard reversible graph2tree transformation).

Another advantage of introducing this tree structure is that the correspondences between strings and abstract trees have been much studied [5, 15, 16]. They can be encoded within the trees by 2 attributes expressing what a node covers lexically (SNODE) and as root of a subtree (STREE).

3. Current status and related research

3.1 Experimental platform

We have implemented a web site called SWIIVRE-UNL[2] (Site on the Web for the Initiation, Information, Validation, Research and Experimentation on UNL [12]) as an experimental basis for our research. It currently allows to:

- get dynamic information on UNL sites,
- access a collection of documents (specs, articles) on UNL,
- browse a collection of aligned sentences and UNL graphs in many languages
- experiment multilingual deconversion,
- try the first version of a Web and XML-oriented UNL graph editor, limited to simple graphs (trees), and programmed using more tags (UNL-xml-ed), DOM, and javaScript [9].

3.2 Building the lattice-tree correspondence

Let us outline the method (currently under implementation) to compute a "best" correspondence. We start with an MS-L0 lattice linked to the text and a UNL-tree produced in a standard way and linked to the UNL graph. The goal is to establish liaisons between the lattice and the tree, and to order the tree so that it is maximally aligned with the lattice, hence with the text. Suppose we have only an L0-English dictionary.

First, we enrich the lattice with English lemmas and the UNL-tree with lemmas of L0, producing MS-L0+EN and UNL-tree+L0. Then, we establish links between nodes of the lattice and of the tree having lemmas in common (in L0 or in English), and compute a score for each trajectory in the lattice. The best trajectory is chosen.

The next phase consists in aligning the tree with that trajectory, using "sure" links as the point of departure, and constraints on the STREE and SNODE liaisons: if there are crossing links, which is possible if two words in the text have similar meanings, preference is given to the link maximizing the proximity in the tree and in the string. Then, liaisons of other types are established:

[2] http://www-clips.imag.fr/geta/User/wang-ju.tsai/welcome.html

lexemes with semantic relations, lexemes with attributes, and MS attributes with attributes.

3.3 Related research

Sending feedback automatically to developers is already done in some MT systems, notably in Taiwan (EKS) and at PAHO [14], but should be much more used than it is. The idea of coedition is also not new: UPM in Madrid uses it to create UNL graphs, Y. Lepage at ATR and Tang E. K. at USM (Penang) have developed editors of string-tree correspondences, Watanabe at IBM-Japan has a very nice interface to edit from a text its underlying dependency structure, the MULTIMETEO system [8] is in effect a coedition system for weather forecasts and their underlying semantic structure, in 6 languages, and there is a project at Xerox working on multilingual generation and free text normalization in restricted domains and typologies (pharmaceutical notices).

In our case, by contrast, coedition is to happen at the consumer side, not (like at UPM) at the producer side, and there is no specific domain or typology. The idea to derive an abstract semantic tree from an IL representation using alignment techniques and not a rule system embedded in a generator seems also to be new.

Conclusion

Coedition of a natural language text and its representation in some interlingual form seems the best way to share text revision across languages. UNL graphs seem to be the best candidates in this context. We have described an approach where, in the simplest sharing scenario, naive users interact directly with the text in their language (L0), and indirectly with the associated graph. It should also be possible to view and directly manipulate the given UNL graph, a lattice or chart produced by some available free morphosyntactic analyzer, and an abstract tree produced not by analysis, but by a standard transformation from the UNL graph, followed by lexical enrichment in L0, and alignment with the text. When completed, our implementation will make it possible to share revision across languages. We will then have progressed towards merging pivot MT, interactive MT, and multilingual text authoring.

References

[1] **Al Assimi A.-B. (2000)** *Gestion de l'évolution non centralisée de documents parallèles multilingues.* Nouvelle thèse, UJF, Grenoble, 31/10/00, 200 p.

[2] **Al Assimi A.-B. & Boitet C. (2001)** *Management of Non-Centralized Evolution of Parallel Multilingual Documents*. Proc. Internationalization Track, 10th International World Wide Web Conference, Hong Kong, May 1-5, 2001, 7 p.

[3] **Blanc E. (2001)** *From graph to tree : Processing UNL graph using an existing MT system.* Proc. First UNL Open Conference - Building Global Knowledge with UNL, Suzhou, China, 18-20 Nov. 2001, UNDL (Geneva), 6 p.

[4] **Boguslavsky I., Frid N., Iomdin L., Kreidlin L., Sagalova I. & Sizov V. (2000)** *Creating a Universal Networking Language Module within an Advanced NLP System.* Proc. COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL & Morgan Kaufmann, H. Uszkoreit ed., pp. 83-89.

[5] **Boitet C. & Zaharin Y. (1988)** *Representation trees and string-tree correspondences.* Proc. COLING-88, Budapest, 22–27 Aug. 1988, ACL, pp. 59—64.

[6] **Boitet C. (1999)** *A research perspective on how to democratize machine translation and translation aids aiming at high quality final output.* Proc. MT Summit VII, Singapore, 13-17 September 1999, Asia Pacific Ass. for MT, J.-I. Tsujii ed., pp. 125—133.

[7] **Boitet C. (2001)** *Four technical and organizational keys for handling more languages and improving quality (on demand) in MT.* Proc. MTS2001 Workshop on "MT2010 — Towards a Road Map for MT", Santiago de Compostela, 18/9/01, IAMT, 8 p.

[8] **Coch J. & Chevreau K. (2001)** *Interactive Multilingual Generation.* Proc. CICLing-2001 (Computational Linguistics and Intelligent Text Processing), Mexico, February 2001, Springer, A. Gelbukh ed., pp. 239-250.

[9] **Jitkue P. (2001)** *Participation au projet SWIIVRE-UNL et première version d'un environnement Web de déconversion multilingue et d'éditeur UNL de base.* Rapport de stage de Maîtrise d'informatique, Université Joseph Fourier, septembre 2001, 13 p.

[10] **Sérasset G. & Boitet C. (1999)** *UNL-French deconversion as transfer & generation from an interlingua with possible quality enhancement through offline human interaction.* Proc. MT Summit VII, Singapore, 13-17 September 1999, Asia Pacific Ass. for MT, J.-I. Tsujii ed., pp. 220—228.

[11] **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. COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL & Morgan Kaufmann, H. Uszkoreit ed., vol. 2/2, pp. 768—774.

[12] **Tsai W.-J. (2001)** *SWIIVRE- a web site for the Initiation, Information, Validation, Research and Experimentation on UNL (Universal Networking Language).* Proc. First UNL Open Conference - Building Global Knowledge with UNL, Suzhou, China, 18-20 Nov. 2001, UNDL (Geneva), 8 p.

[13] **Uchida H. (1989)** *ATLAS.* Proc. MTS-II (MT Summit), Munich, 16-18 août 1989, pp. 152-157.

[14] **Vasconcellos M. & León M. (1988)** *SPANAM and ENGSPAN : Machine Translation at the Pan American Health Organization.* In "Machine Translation systems", J. Slocum, ed., Cambridge Univ. Press, pp. 187—236.

[15] **Vauquois B. & Chappuy S. (1985)** *Static grammars: a formalism for the description of linguistic models.* Proc. TMI-85 (Conf. on theoretical and metholodogical issues in the Machine Translation of natural languages), Aug. 1985, pp. 298-322.

[16] **Zaharin Y. (1986)** *Strategies and heuristics in the analysis of a natural language in Machine Translation.* Proc. COLING-86, Bonn, Aug. 1986, pp. 136—139.