Computer Science Journal of Moldova, vol.**14**, no.1(40), 2006

From Word Alignment to Word Senses, via Multilingual Wordnets

Dan Tufiş

Abstract

Most of the successful commercial applications in language processing (text and/or speech) dispense with any explicit concern on semantics, with the usual motivations stemming from the computational high costs required for dealing with semantics, in case of large volumes of data. With recent advances in corpus linguistics and statistical-based methods in NLP, revealing useful semantic features of linguistic data is becoming cheaper and cheaper and the accuracy of this process is steadily improving. Lately, there seems to be a growing acceptance of the idea that multilingual lexical ontologies might be the key towards aligning different views on the semantic atomic units to be used in characterizing the general meaning of various and multilingual documents. Depending on the granularity at which semantic distinctions are necessary, the accuracy of the basic semantic processing (such as word sense disambiguation) can be very high with relatively low complexity computing. The paper substantiates this statement by presenting a statistical/based system for word alignment and word sense disambiguation in parallel corpora. We describe a word alignment platform which ensures text pre-processing (tokenization, POS-tagging, lemmatization, chunking, sentence and word alignment) as required by an accurate word sense disambiguation.

1 The Pervasive Ambiguity of Natural Languages

Most difficult problems in natural language processing stem from the inherent ambiguous nature of the human languages. Ambiguity is

present at all levels of traditional structuring of a language system (phonology, morphology, lexicon, syntax, semantics) and not dealing with it at the proper level, exponentially increases the complexity of the problem solving. Currently, the state of the art taggers (combining various models, strategies and processing tiers) ensure no less than 97-98% accuracy in the process of morpho-lexical full disambiguation. For such taggers a 2-best tagging[1] is practically 100% correct.

One further step is the word sense disambiguation (WSD) process. In the traditional compositional semantics, the meaning of a complex expression is supposed to be derivable from the meanings of its parts, and the way in which those parts are combined. Depending on the representation formalisms for the word-meaning representation, various calculi may be considered for computing the meaning of a complex expression from the atomic representations of the word senses. Obviously, one should be able, before hand, to decide for each word in a text which of its possible meanings is, contextually, the right one.

Therefore, it is a generally accepted idea that the WSD task is highly instrumental (if not indispensable) in semantic processing of natural language documents.

It is almost a truism that more decision makers, working together, are likely to find a common solution superior to each solution individually found. Dieterich [1] discusses conditions under which different decisions (in his case classifications) may be combined for obtaining a better result. Essentially, a successful automatic combination method would require comparable performance on behalf of the decision makers and, additionally, that they would not make the similar errors. This idea has been exploited by various NLP researchers in language modelling, statistical POS tagging, parsing, word alignment, word sense disambiguation, etc.

The WSD problem can be stated as being able to associate to an ambiguous word ($\boldsymbol{w}$) in a text or discourse, the sense ($\boldsymbol{s}_k$) which is dis-

[1]In k-best tagging, instead of assigning each word exactly one tag (the most probable in the given context), it is allowed to have occasionally at most k-best tags attached to a word and if the correct tag is among the k-best tags, the annotation is considered to be correct.

tinguishable from other senses (s_1, ..., s_{k-1}, s_{k+1}, ..., s_n) prescribed for that word by a reference semantic lexicon. One such semantic lexicon (actually a lexical ontology) is Princeton WordNet [2] version 2.0[2] (henceforth PWN). PWN is a very fine-grained semantic lexicon currently containing 203,147 sense distinctions, clustered in 115,424 equivalence classes (synsets). Out of the 145,627 distinct words, 119,528 have only one single sense. However, the remaining 26,099 words are those that one would frequently meet in a regular text and their ambiguity ranges from two senses up to 36. Several authors considered that sense granularity in PWN is too fine-grained for the computer use, arguing that even for a human (native speaker of English) the sense differences of some words are very hard to be reliably (and systematically) distinguished. There are several attempts to group the senses of the words in PWN in coarser grained senses – *hyper-senses* – so that clear-cut distinction among them is always possible for humans and (especially) computers. We will refer in this paper to two hyper-sense inventories used in the BalkaNet project [3]. A comprehensive review of the WSD state-of the art at the end of 90's can be found in [4]. Stevenson and Wilks [5] review several WSD systems that combined various knowledge sources to improve the disambiguation accuracy and address the issue of different granularities of the sense inventories. SENSEVAL[3] series of evaluation competitions on WSD is a very good source on learning how WSD evolved in the last 6-7 years and where is it nowadays.

We describe a multilingual environment, containing several monolingual wordnets, aligned to PWN used as an interlingual index (ILI). The word-sense disambiguation method combines word alignment technologies, and interlingual equivalence relations in multilingual wordnets [6]. Irrespective of the languages in the multilingual documents, the words of interest are disambiguated by using the same sense-inventory labels. The aligned wordnets were constructed in the context of the European project BalkaNet. The consortium developed monolingual wordnets for five Balkan languages (Bulgarian, Greek, Romanian Serbian, and Turkish) and extended the Czech wordnet initially developed

[2] http://www.cogsci.princeton.edu/~wn/
[3] http://www.cs.unt.edu/~rada/senseval

in the EuroWordNet project [6]. The wordnets are aligned to PWN, taken as an interlingual index, following the principles established by the EuroWordNet consortium. The version of the PWN used as ILI is an enhanced XML version where each synset is mapped onto one or more SUMO [7] conceptual categories and is classified under one of the IRST domains [8]. In the present version of the BalkaNet ILI there are used 2066 SUMO distinct categories and 163 domain labels. Therefore, for our WSD experiments we had at our disposal three sense-inventories, with very different granularities: PWN senses, SUMO categories and IRST Domains.

2 Word Alignment

The word alignment is the first step (the hardest) in our approach for the identification of word senses. In order to reduce the search space and to filter out significant information noise, the context is reduced to the level of sentence. Therefore, a parallel text $< T_{L1}T_{L2} >$ is represented as a sequence of pairs of one or more sentences in language L1 (S^1_{L1} S^2_{L1}...S^k_{L1}) and one or more sentences in language L2 (S^1_{L2} S^2_{L2}... S^m_{L2}) so that the two ordered sets of sentences represent reciprocal translations. Such a pair is called a translation alignment unit (or translation unit). The word alignment of a bitext is an explicit representation of the pairs of words $< w_{L1}w_{L2} >$ (called translation equivalence pairs) co-occurring in the same translation units and representing mutual translations. The general word alignment problem includes the cases where words in one part of the bitext are not translated in the other part (these are called *null alignments*) and the cases where multiple words in one part of the bitext are translated as one or more words in the other part (these are called expression alignments).

The input format is obtained from two raw texts that represent reciprocal translations. If not already sentence aligned, the two texts are aligned by a sentence aligner, similar to Moore's aligner [9] but which unlike it, is able to recover the non-one-to-one sentence alignments. The texts in each language are then tokenized, tagged and lemmatized. Frequently, the translation equivalents have the same part-of

speech, but relying on such a restriction would seriously affect the alignment recall. However, when the translation equivalents have different parts of speech, this difference is not arbitrary. *POS affinities,* $\{p(POS_m^{L1}|POS_n^{L2}), p(POS_n^{L2}|POS_m^{L1})\}$, are easy to estimate and we use them to filter out improbable translation equivalents pairs.

The next pre-processing step is represented by the sentence chunking in both languages. The chunks are recognized by a set of regular expressions defined over the tagsets and they correspond to (non-recursive) noun phrases, adjectival phrases, prepositional phrases and verb complexes (analytical realization of tense, aspect mood and diathesis and phrasal verbs). The texts are further processed by a statistical dependency linking parser. Finally, the bitext is assembled as an XML document (XCES[4] compliant format), which is the standard input for most of our tools.

2.1 Two Aligners and Their Combination

We developed two quite different word aligners, motivated by two distinct objectives: the first one, called YAWA, was motivated by a project aiming at the development of an interlingually aligned set of wordnets while the other one was developed within an SMT ongoing project. The first one was used for validating, against a multilingual corpus, the interlingual synset equivalences and also for WSD experiments. Although, initially it was concerned only with open class words recorded in a wordnet, turning it into an "all words" aligner was not a difficult task. YAWA is a three stage lexical aligner that uses bilingual translation lexicons and phrase boundaries detection to align words of a given bitext. The translations lexicons are generated by a different module, TREQ [10], which generates translation equivalence hypotheses for the pairs of words (one for each language in the parallel corpus) which have been observed occurring in aligned sentences more than expected by chance. The hypotheses are filtered by a loglikelihood score threshold. Several heuristics (string similarity-cognates, POS affinities and

[4] http://www.cs.vassar.edu/XCES/

alignments locality[5]) are used in a competitive linking manner [11] to extract the most likely translation equivalents.

YAWA generates a bitext alignment by incrementally adding new links to those created at the end of the previous stage. The existing links act as contextual restrictors for the new added links. From one phase to the other, new links are added without deleting anything. This monotonic process requires a very high precision (at the price of a modest recall) for the first step. The next two steps are responsible for significantly improving the recall and ensuring an increased F-measure.

In the rest of this section we present in some details the various steps of the two aligners, evaluate them individually and finally describe the combination of the alignments produced by YAWA and MEBA and evaluate the result of the combination.

2.2 YAWA

2.2.1 YAWA Phase 1: Content Words Alignment

YAWA begins by taking into account only very probable links that will represent the skeleton alignment to be the input for the second phase. This alignment is done using outside resources such as translation lexicons and involves only the alignment of content words (nouns, verbs, adjective and adverbs).

The bitext to be word-aligned is concatenated to a reference parallel corpus containing the languages of concern. For Romanian-English we use an almost 1.5 million words parallel corpus. The concatenation of the target bitext to the reference corpus is required in case the target bitext is too small to provide reliable statistical evidence for the possible translation equivalents that are extracted by the TREQ module. The translation equivalence pairs are ranked according to an association score (i.e. log-likelihood, DICE, point-wise mutual information, etc.).

[5]The *alignments locality* heuristics exploits the observation made by several researchers that adjacent words of a text in the source language tend to align to adjacent words in the target language. A more strict alignment locality constraint requires that all alignment links starting from a chunk, in the one language end in a chunk in the other language.

We found that the best filtering of the translation equivalents was the one based on the log-likelihood (LL) score with a threshold of 9.

Each translation unit (pair of aligned sentences) of the parallel corpus is scanned for establishing the most likely links based on a competitive linking strategy that takes into account the LL association scores given by the TREQ translation lexicon. If a candidate pair of words is not found in the translation lexicon, we compute their orthographic similarity (cognate score [10]). If this score is above a predetermined threshold (we used the empirically found value of 0.43), the two words are treated as if they existed in the translation lexicon with a high association score (in practice we have multiplied the cognate score by 100 to yield association scores in the range 0..100).

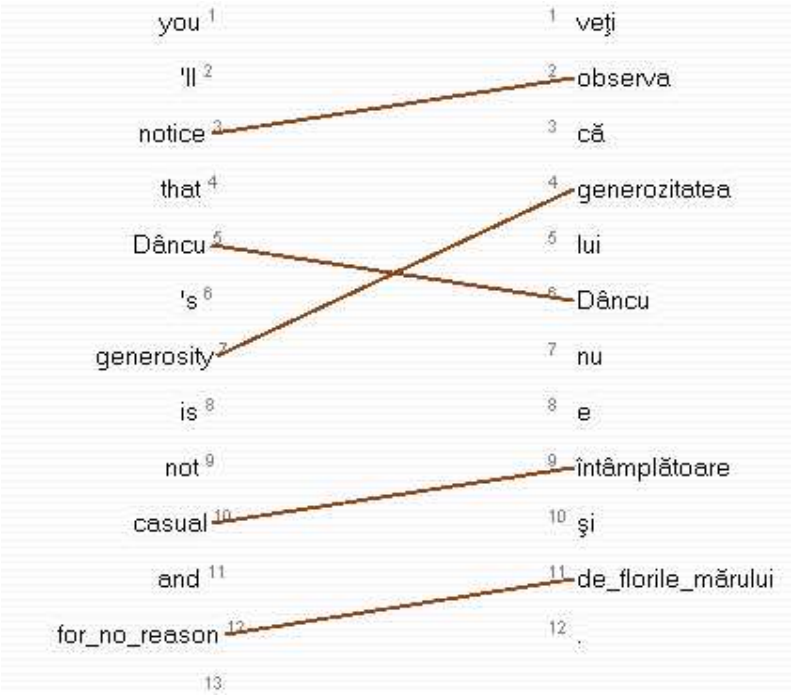


Figure 1. Alignment after the first step

The Figure 1 exemplifies the links created between two tokens of a parallel sentence by the end of the first phase.

2.2.2 YAWA Phase 2: Chunks Alignment

The second phase requires chunking of each part of the bitext. In our Romanian-English experiments, this requirement was fulfilled by using a set of regular expressions defined over the tagsets used in the target bitext. These simple chunkers recognize noun phrases, prepositional phrases, verbal and adjectival or adverbial groupings of both languages.

In this second phase YAWA firstly achieve the chunk-to-chunk matching and after that, continues with aligning the words of aligned chunks. Chunk alignment is done on the basis of the skeleton alignment produced in the first phase. The algorithm is simple: align two chunks $c(i)$ in source language and $c(j)$ in the target language if $c(i)$ and $c(j)$ have the same type (noun phrase, prepositional phrase, verbal group, adjectival/adverbial group) and if there exists a link $\langle w(s), w(t)\rangle$ so that $w(s) \in c(i)$ then $w(t) \in c(j)$.

After the chunks were aligned, a language pair dependent module takes over to align the unaligned words belonging to the chunks. Our module for the Romanian-English pair of languages contains some very simple empirical rules such as: if b is aligned to c and b is preceded by a, link a to c, unless there exist d in the same chunk with c and the POS category of d has a significant affinity with the category of a. The simplicity of these rules derives from the shallow structures of the chunks. In the above example b and c are content words while a is very likely a determiner or a modifier for b. The result of the second alignment phase, considering the same sentence from Figure 1, is exemplified in Figure 2. The new links are represented by the double lines:

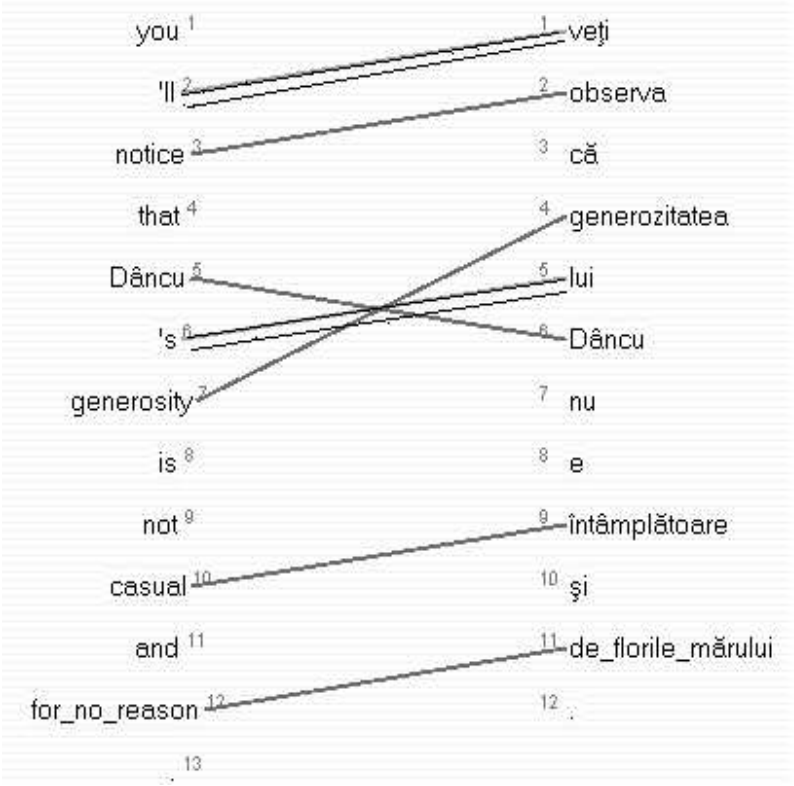


Figure 2. Alignment after the second step

2.2.3 YAWA Phase 3: Remaining Blocks of Words Alignment

This phase identifies contiguous sequences of words in each part of the bitext which remain unaligned and try to heuristically align the words of the best matching such blocks. The main criteria used is the POS-affinities of the remaining unaligned words and their relative positions. Let us exemplify, using the same example and the result shown in Figure 2, the way of adding new links in this last phase of the alignment. At the end of phase 2 the blocks of consecutive words

that remain to be aligned are: English {en_1 = (that), en_2 = (is, not), en_3 = (and), en_4 = (.)} and Romanian {ro_1 = (că), ro_2 = (nu, e), ro_3 = (şi), ro_4 = (.)}. The mapping of source and target unaligned blocks depends on two criteria: the surrounding chunks are already aligned, and the pairs in the candidate unaligned blocks have significant POS-affinities. For instance in the Figure 2, blocks en_1 = (that) and ro_1 = (că) satisfy the above conditions because they appear among already aligned chunks (<'ll notice> ⇔ <veţi observa> and <Dâncu 's generosity> ⇔ <generozitatea lui Dâncu>) and they contain words with the same POS.

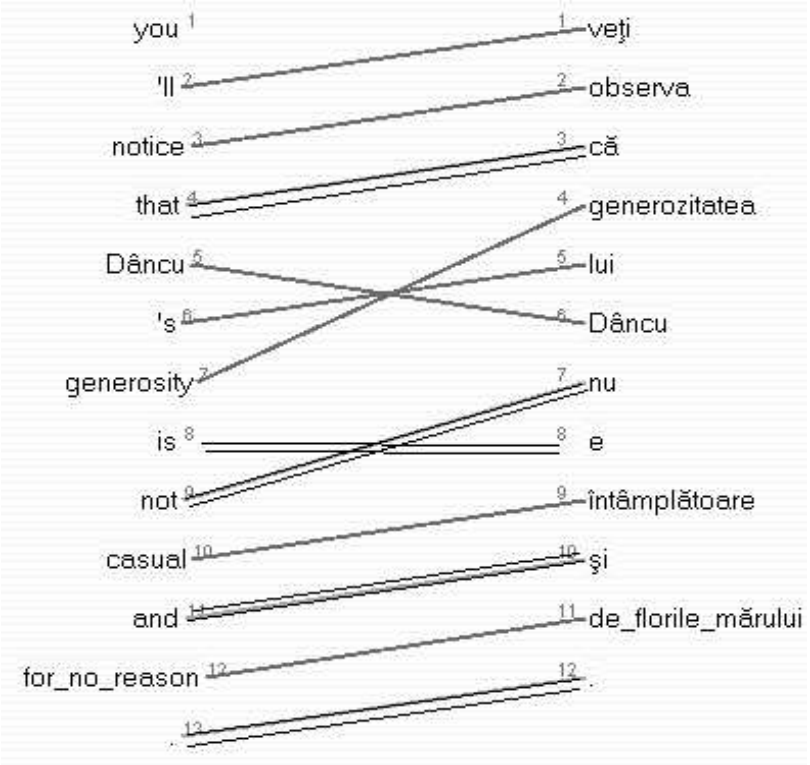


Figure 3. Alignment after the third step

After block alignment[6], given a pair of aligned blocks, the algorithm links words with the same POS and then the phase 2 is called again with these new links as the skeleton alignment. In Figure 3 is shown the result of phase 3 alignment of the sentence we used as an example throughout this section. The new links are shown (as before) by double lines.

The third phase is responsible for significant improvement of the alignment recall, but it also generates several wrong links. The detection of some of them is quite straightforward, and we added an additional correction phase 3.f. Romanian being a relatively free order language, it is quite easy to produce good quality translation by preserving the order of most of phrasal groups. We noticed this tendency in most of our training bilingual data so, we used this finding as an additional filter to remove those links that cross several regularly distributed links along the alignment.

2.2.4 YAWA Performance Analysis

The Table 1 presents the results of the YAWA aligner at the end of each alignment phase. Although the Precision decreases from one phase to the next one, the Recall gains are significantly higher so, the F-measure is monotonically increasing.

Table 1. YAWA evaluation

	Precision (P)	Recall (R)	F-Measure (F)
Phase 1	94.08%	34.99%	51.00%
Phase 1+2	89.90%	53.90%	67.40%
Phase 1+2+3	88.82%	73.44%	80.40%
Phase 1+2+3+3.f	**88.80%**	**74.83%**	**81.22%**

[6] Only 1:1 links are generated between blocks.

2.3 MEBA

2.3.1 MEBA Reifying Aligner

A quite different approach from the one used by YAWA, is implemented in our second word aligner, called **MEBA**. It is a multiple parameter and multiple step algorithm using relevance thresholds specific to each parameter, but different from each step to the other. The implementation of MEBA was strongly influenced by the famous five IBM models described in the [12] seminal paper. We used GIZA++ [13, 14] to estimate different parameters of the MEBA aligner.

MEBA is an iterative algorithm that takes advantage of all preprocessing phases mentioned in the beginning of the Section 2.

The alignment model considers a link between two candidate words as an object that is described by a feature-values structure (with values in the [0,1] interval) which we call the *reification* of the link. We differentiate between *context independent features* that refer only to the tokens of the current link (translation equivalency, part-of-speech affinity, cognates, etc.) and *context dependent features* that refer to the properties of the current link with respect to the rest of links in a bi-text (locality, number of traversed links, tokens indexes displacement, collocation). Also, we distinguish between bi-directional features (translation equivalency, part-of-speech affinity) and non-directional features (cognates, locality, number of traversed links, collocation, indexes displacement)

The program starts building the most probable links *(anchor links)*: cognates, numbers, dates, and translation pairs with high translation probabilities. Then, it iteratively aligns content words (open class categories) in the immediate vicinity of the anchor links. The links to be added at any later step are supported or restricted by the links created in the previous iterations. Each of the iterations can be configured to align different categories of tokens (named entities, dates and numbers, content words, functional words, punctuation) in decreasing order of statistical evidence, with different weights and different significance thresholds on each feature and iteration.

The score of a candidate link (LS) between a source token i and a

target token j is computed by a linear function of the features scores:

$$LS(i,j) = \sum_{i=1}^{n} \lambda_i * ScoreFeat_i; \sum_{i=1}^{n} \lambda_i = 1$$

In the following subsection we briefly discuss the main features we use in reifying a link.

2.3.2 MEBA Features

In this section we will denote by A and B the source and target lexical items respectively.

Translation equivalence. The word aligner invokes GIZA++ to build translation probability lists for either lemmas or the word-forms of the bitext. The considered token for the translation model build by GIZA++ is the respective lexical item (lemma or word-form) trailed by its POS tag (eg. plane_N, plane_V plane_A). In this way we avoid data sparseness and filter noisy data. A further way of removing the noise created by GIZA++ is to filter out all the translation pairs below a LL-threshold. We made various experiments and empirically set the value of this threshold to 6. All the probability losses by this filtering were redistributed proportionally to their initial probabilities to the surviving translation equivalence candidates.

Translation equivalence entropy score. The translation equivalence entropy score is a favouring parameter for the words with a skewed probability distribution for their translation equivalents[7]. Since this feature is definitely sensitive to the order of the lexical items, we compute an average value for the link: αES(A)+βES(B). Currently we use $\alpha = \beta = 0.5$, but it might be interesting to see, depending on different language pairs, how the performance of the aligner would be affected by different settings of these parameters.

$$ES(W) = 1 - \frac{-\sum_{i=1}^{N} p(W,TR_i) * \log p(W,TR_i)}{\log N}$$

[7]This heuristics implements the zipffian conjecture about the word senses distribution in a coherent text.

Part-of-speech affinity. In faithful translations the translated words tend to be translated by words of the same part-of-speech. When this is not the case, the different POSes, are not arbitrary. The part of speech affinity, P(POS(A)|POS(B)), can be easily computed from a gold standard alignment. Obviously, this is a directional feature, so an interpolation operation is necessary in order to ascribe this feature to a link:

$$\text{PA}=\alpha\ \text{P(POS(A)|POS(B))} + \beta\ \text{P(POS(A)|POS(B))}.$$

Again, we used $\alpha = \beta = 0.5$ but different values of these weights might be worthwhile investigating.

Cognates. The similarity measure, COGN(A, B), is implemented as a Levenstein metric. Using the COGN test as a filtering device is a heuristic based on the *cognate conjecture* which says that when the two tokens of a translation pair are orthographically similar, they are very likely to have similar meanings (i.e. they are cognates). The threshold for the COGN(A, B) test was empirically set to 0.43. This value depends on the pair of languages in the bitext. The actual implementation of the COGN test includes a language-dependent normalisation step, which strips some suffixes, discards the diacritics, reduces some consonant doubling, etc. This normalisation step was hand written, but, based on available lists of cognates, it could be automatically induced.

Obliqueness. Each token in both sides of a bitext is characterized by a position index, computed as the ratio between the relative position in the sentence and the length of the sentence. The absolute value of the difference between tokens' position indexes, subtracted from 1 gives the link's "obliqueness".

$$OBL(A_i, B_j) = 1 - \left| \frac{i}{length(Sent_S)} - \frac{j}{length(Sent_T)} \right|$$

Locality is a feature that estimates the degree to which the links are sticking together.

MEBA has three features to account for locality: (i) *weak locality*, (ii) *chunk-based locality* and (iii) *dependency-based* locality (see Figure 4).

The value of the *weak locality* feature is derived from the already existing alignments in a window of N tokens centred on the focused token. The window size is variable, proportional to the sentence length. If in the window there exist k linked tokens and the indexes of their links are $< i_1 j_1 >, \ldots < i_k j_k >$ then the locality feature of the new link $< i_{k+1}, j_{k+1} >$ is defined by the equation below:

$$LOC = \min(1, \frac{1}{k} \sum_{m=1}^{k} \frac{|i_{k+1} - i_m|}{|j_{k+1} - j_m|}).$$

In the case of *chunk-based locality* the window span is given by the indexes of the first and last tokens of the chunk.

Dependency-based locality uses the set of the dependency links of the tokens in a candidate link for the computation of the feature value. In this case, the LOC feature of a candidate link $< i_{k+1}, j_{k+1} >$ is set to 1 or 0 according to the following rule:

> if between i_{k+1} and i_α there is a (source language) dependency and if between j_{k+1} and j_β there is also a (target language) dependency then LOC is 1 if i_α and j_β are aligned, and 0 otherwise.
>
> Note that in case $j_{k+1} \equiv j_\beta$ a trivial dependency (identity) is considered and the LOC attribute of the link $< i_{k+1}, j_{k+1} >$ is set always to 1.

Collocation. We used the bi-grams list to annotate the chains of lexical dependencies among the contents words. Then, the value of the collocation feature is computed similar to the dependency-based locality feature. The algorithm searches for the links of the lexical dependencies around the candidate link.

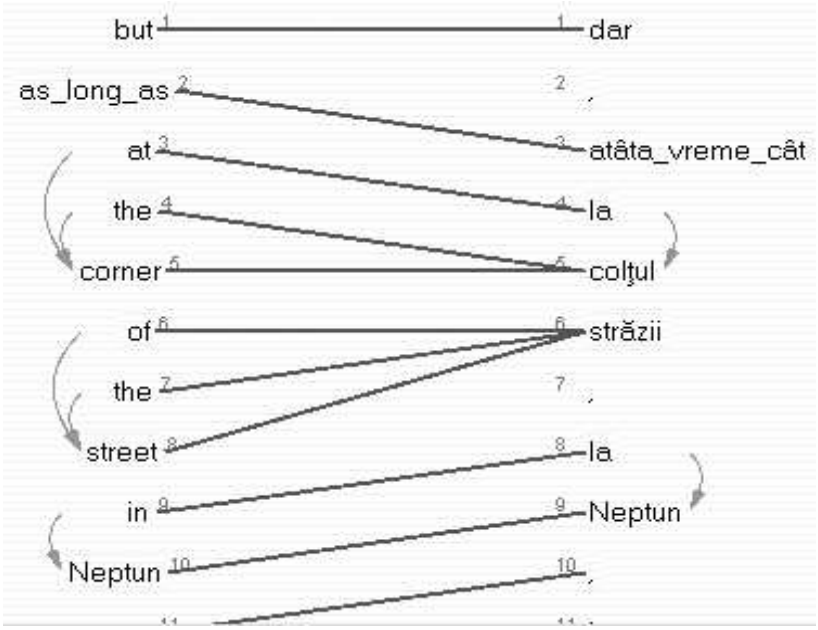


Figure 4. Chunk and dependency-based locality

Monolingual collocation is an important clue for word alignment. If a source collocation is translated by a multiword sequence, very often the lexical cohesion of source words can also be found in the corresponding translated words. In this case the aligner has strong evidence for many to many linking. When a source collocation is translated as a single word, this feature is a strong indication for a many to 1 linking.

Bi-gram lists (only content words) were built from each monolingual part of the training corpus, using the log-likelihood score (threshold of 10) and minimal occurrence frequency (3) for candidates filtering.

We used the bi-grams list to annotate the chains of lexical dependencies among the contents words (see Figure 5). Then, the value of the collocation feature is computed similar to the dependency-based locality feature. The algorithm searches for the links of the lexical dependencies around the candidate link.

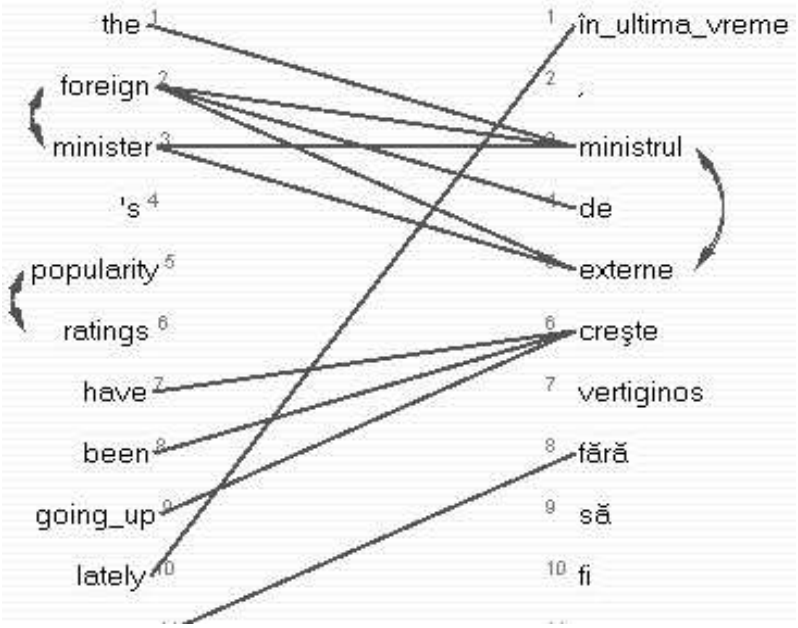


Figure 5. Collocation feature

2.3.3 MEBA Performance Analysis

The cumulative results of the major processing steps are shown in the table below. As one can see the precision decreases from the first step to the last with 6.25% but the recall (almost three times better) and the F-measure (almost double) are significantly improved.

The alignments generated by MEBA were compared to the ones produced by YAWA and evaluated against the Gold Standard (GS) annotations used in the Word Alignment Shared Tasks (Romanian-English track) organized at HLT-NAACL2003 [15].

As one can observe from the results in Table 1 and Table 2 the two aligners, which are based on quite different models, have comparable performances. Moreover, by analyzing the alignment errors done by

Table 2. MEBA evaluation

	Precision	Recall	F-measure
"Anchor" links	98.40%	28.82%	44.58%
Words around "anchors"	96.28%	44.32%	60.70%
Functional words and punctuation	93.23%	61.98%	74.46%
Probable links	**92.15%**	**73.40%**	**81.71%**

each word aligner, we found that the number of common mistake was small so, the premises for a successful combination were very good [1].

2.4 COWAL: The Combined Aligner

The Combined Word Aligner, **COWAL**, is a wrapper of the two aligners (YAWA and MEBA) merging the individual alignments and filtering the result. At the Shared Task on Word Alignment organized by the ACL2005 Workshop on "Building and Using Parallel Corpora: Data-driven Machine Translation and Beyond" [16], we participated (on the Romanian-English track) with the two aligners and the combined one (COWAL). Out of 37 competing systems, COWAL [17] was rated the first, MEBA the 20^{th} and TREQ-AL [18], (the former version of YAWA), was rated the 21^{st}. The usefulness of the aligner combination was convincingly demonstrated. Meanwhile, both the individual aligners were significantly improved as well as their combination.

One very simple, but very effective method of alignment combination is a heuristic procedure which merges the alignments produced by two or more word aligners and filters out the links that are likely to be wrong. For the purpose of filtering, a link is characterized by its type defined by the pair of indexes (i,j) and the POS of the tokens of the respective link. The likelihood of a link is proportional to the POS affinities of the tokens of the link and inverse proportional to the *bounded relative positions* (BRP) of the respective tokens: $BRP = 1 + ||i - j| - avg|$ where avg is the average displacement in a Gold Standard of the aligned tokens with the same POSes as the

tokens of the current link. From the same gold standard we estimated a threshold below which a link is removed from the final alignment.

A more elaborated alignment combination (with better results than the previous one) is modelled as a binary statistical classification problem (good / bad) and, as in the case of the previous method, the net result is the removal of the links which are likely to be wrong. We used the SVM training and classification toolkit - LIBSVM [19] with the default parameters (C-SVC classification and radial basis kernel function). Both context independent and context dependent features characterizing the links were used for training. The classifier was trained with positive and negative examples of links. A subset of the Gold Standard alignment links was used as positive examples set. The same number of negative examples was extracted from the alignments produced by COWAL and MEBA where they differ from the Gold Standard.

The result of the SVM-based combination (COWAL), compared with the individual aligners, is shown in Table 3.

Table 3. Combined alignment

Aligner	P	R	F-measure
YAWA	88.80%	74.83%	81.22%
MEBA	92.15%	73.40%	81.71%
COWAL	87.26%	80.94%	83.98%

COWAL is now embedded into a larger platform (called MTkit) that incorporates the tools for bitexts pre-processing, a graphical interface that allows for comparing and editing different alignments, as well as a word sense disambiguation module. A snapshot of the COWAL graphical interface is shown in Figure 6. The left pane in Figure 6 is the alignment viewer and editor area. The user can edit the alignments (delete and add one or multiple links). By double clicking a word in this pane, its properties will be automatically displayed in the right-hand windows. The upper-right window shows the lexico-syntactic

properties of the selected word: the morphological analysis of the orthographic form, its lemma, the syntactic chunk to which it belongs. Currently this pane is not editable. The bottom-right window displays the semantic properties of the selected word: its sense in the current context, the gloss for this sense, synonyms, hyperonyms, derivatives, etc. These properties are extracted from the wordnet of the language to which the selected word belongs to. This pane is editable, but only the sense number is subject to user modifications.

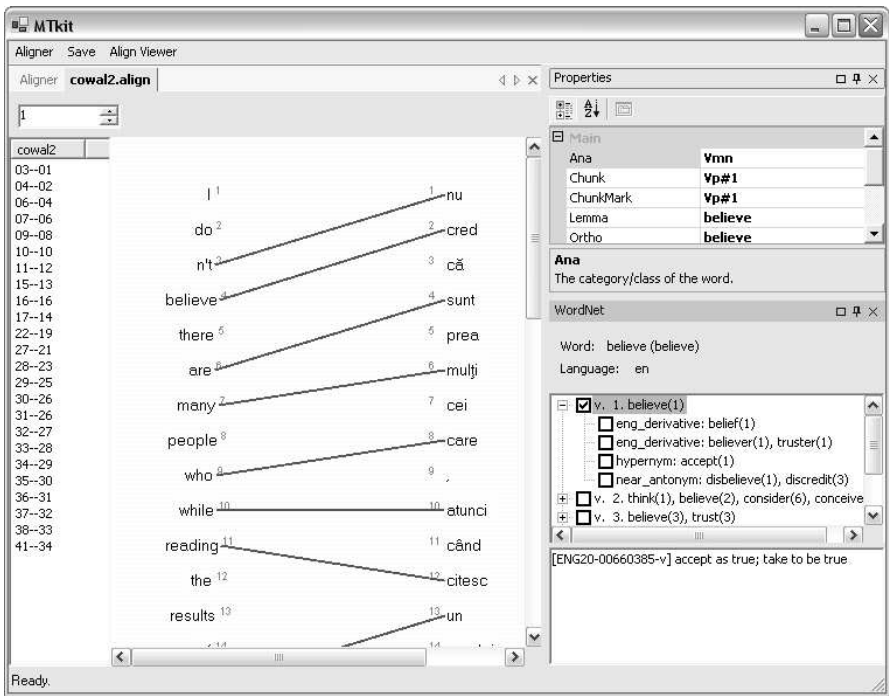


Figure 6. COWAL Graphical User Interface

Although far from being perfect, the accuracy of word alignments and of the translation lexicons extracted from parallel corpora is rapidly

improving. In the shared task evaluations of different word aligners, organized on the occasion of the 2003 NAACL Conference and the 2005 ACL Conference, our winning systems TREQ-AL [18] and COWAL [17] produced wordnet-relevant lexicons[8] with F-measures as high as 84.26% and 89.92%[9].

3 Wordnet-based Sense Disambiguation

The task of word sense disambiguation (WSD) requires one reference sense inventory in terms of which the senses of the target words will be labeled. We argued at length elsewhere [20] that a meaningful discussion of the performances of a WSD system cannot dispense of clearly specifying the sense inventory it uses, and the comparison between two WSD systems that uses different sense inventories is frequently more confusing than illuminating. Essentially, this is because the differences in the semantic distinctions (sense granularities), as used by different semantic dictionaries (sense repositories), make the difficulty of the WSD task range over a large spectrum. For instance, the discrimination of homographs (more often than not having different parts of speech, e.g. "(to) bottle" as storing liquids or gases in bottles, versus "bottle" as the recipient) is much simpler than metonymic distinctions (e.g. "bottle" as container, versus "bottle" as content).

In our research, we used the Princeton Wordnet 2.0 as the major sense inventory and the BalkaNet multilingual lexical ontology. The BalkaNet lexical ontology has been developed within the European project with the same name (September 2001-August 2004) and includes five languages from the Balkan area (Bulgarian, Greek, Romanian, Serbian and Turkish), plus Czech, the wordnet of which, initially developed in EuroWordNet, has been significantly extended. By observing the interlingual synset mapping principle and incorporating most of the conceptual extensions proposed by EuroWordNet, the

[8] wordnet-relevant lexicons are restricted only to translation pairs of the same major POS (nouns, verbs, adjectives and adverbs).

[9] Currently, with the most recent improvements, COWAL's F-measure is 92.08%

BalkaNet wordnets can be easily combined with any of the other semantic networks of the EuroWordnet, and, thus, one may speak about a really pan-European multilingual lexical ontology, covering at least 15 languages[10].

The BalkaNet multilingual environment took advantage of the latest developments in the PWN that was adopted itself as an interlingual index. This is a major difference with respect to the EuroWordNet's ILI. As the SUMO/MILO [7] and DOMAINS [8], have been aligned with PWN, they automatically became available in each monolingual wordnet of the BalkaNet. To allow the representation of language idiosyncratic properties, structural knowledge present in the monolingual wordnets has precedence over the structural knowledge imported from the ILI. As the Romanian wordnet imported SUMO/MILO and DOMAINS labels and the synsets unique identifiers are the same as in the PWN, it is self-contained but at the same time unambiguously integratable in a PWN centered multilingual wordnet infrastructure.

Once the translation equivalents identified, it is reasonable to expect that the words of a translation pair $< w^i_{L1}, w^j_{L2} >$ share at least one conceptual meaning stored in an interlingual sense inventory. When interlingually aligned wordnets are available (as is our case), obtaining the sense labels for the words in a translation pair is straightforward: one has to identify for w^i_{L1} the synset S^i_{L1} and for w^j_{L2} the synset S^j_{L2} so that S^i_{L1} and S^j_{L2} are projected over the same interlingual concept. The index of this common interlingual concept (ILI) is the sense label of the two words w^i_{L1} and w^j_{L2}. However, it is possible that no common interlingual projection will be found for the synsets to which w^i_{L1} and w^j_{L2} belong. In this case, the senses of the two words will be given by the indexes of the most similar interlingual concepts corresponding to the synsets of the two words. Our measure of interlingual concepts semantic similarity is based on PWN structure. We compute the semantic-similarity[11] score by the formula $SYM(ILI_1, ILI_2) = \frac{1}{1+k}$

[10]Basque, Bulgarian, Catalan, Dutch, Czech, English, Estonian, French, German, Greek, Italian, Romanian, Serbian, Spanish, and Turkish.

[11]For a detailed discussion and an in-depth analysis of several other measures see: Budanitsky, A., Hirst, G., Semantic distance in WordNet: An experimental,

where k is the number of links from ILI_1 to ILI_2 or from both ILI_1 and ILI_2 to the nearest common ancestor. In Figure 7 and Figure 8, we exemplify the process of sense labeling of the words in two translation pairs as detected by the word alignment phase.

Let us consider first the pair <**lamp, lampă**>. Looking up the English and Romanian wordnets for the synsets that contain the words "lamp" and "lampă" respectively, we find the following lists of unique identifiers that differentiate among the noun senses of the two words:

PWN2.0 (lamp) = {03500372-n, 03500773-n}

RoWN (lampă) = {03500773-n, 03500872-n}

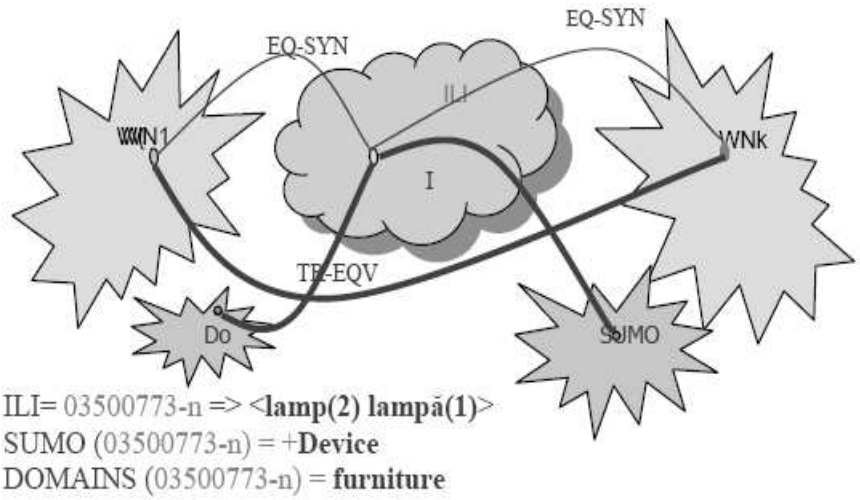


Figure 7. <**lamp lampă**>

The intersection reveals one common identifier (03500773-n) which, therefore, is taken as the common interlingual meaning. From an ILI

application-oriented evaluation of five measures. Proceedings of the Workshop on WordNet and Other Lexical Resources, NAACL, Pittsburgh, June, (2001) 29-34.

code, one can deterministically determine the SUMO concept and the DOMAINS label (see Figure 7).

Now, if we consider a different translation equivalent for the word **"lamp"**, namely **"felinar"** and repeat the procedure described above,

PWN2.0 (lamp) = {03500372-n, 03500773-n}

RoWN (felinar) = {003505057-n}

we notice that there is no common interlingual ILI code in the two lists. In this case, the metrics mentioned above is used to select the closest related senses: SYM(03500372-n,003505057-n)=0.5; SYM(03500373-n,003505057-n)=0.125 (see Figure 8).

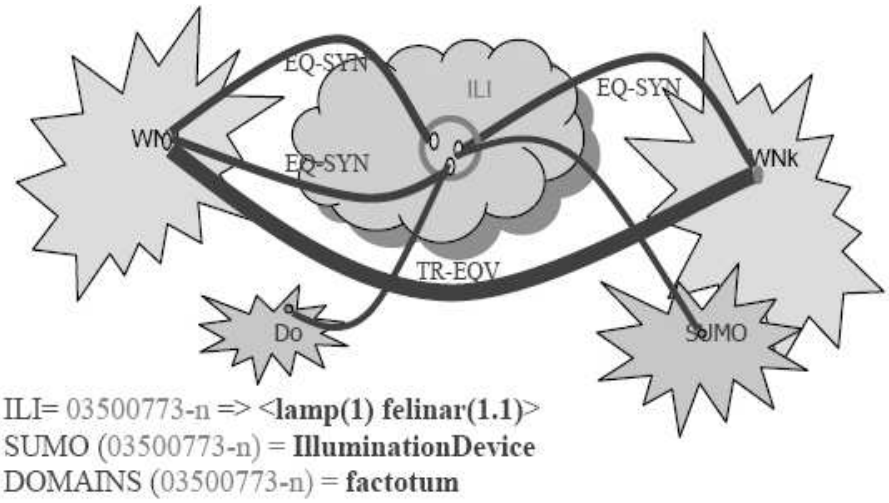


Figure 8. <**lamp felinar**>

After the WSD process has finished, the sense information is inserted into the XML encoding of the corpus. Which sense inventory (ILI, SUMO or DOMAINS) should be used in the encoding is a user-set parameter, which by default includes all of them.

In Figure 9, it is shown the final encoding of one translation unit of the "1984" parallel corpus. The "sn" attribute represents the Princeton Wordnet 2.0 unique synset identifier (ILI code), the "oc" attribute represents the SUMO ontology concept and the "dom" attribute represents the DOMAINS label.

```
<tu id="Ozz20">
 <seg lang="en">
 <s id="Oen.1.1.4.9">
  <w lemma="the" ana="Dd">The</w>
  <w lemma="patrol" ana="Ncnp" sn="3" oc="Group" dom="military">patrols</w>
  <w lemma="do" ana="Vais">did</w>
  <w lemma="not" ana="Rmp" sn="1" oc="not" dom="factotum">not</w>
  <w lemma="matter" ana="Vmn" sn="1" oc="SubjAssesAttr" dom="factotum">matter</w>
  <c>,</c>
  <w lemma="however" ana="Rmp" sn="1" oc="SubjAssesAttr|PastFn" dom="factotum">however</w>
  <c>.</c>
 </s>
</seg>
<seg lang="ro">
 <s id="Oro.1.2.5.9">
  <w lemma="şi" ana=Crssp>Şi</w>
  <w lemma="totuşi" ana="Rgp" sn="1" oc="SubjAssesAttr|PastFn" dom="factotum">totuşi</w>
<c>,</c>
  <w lemma="patrulă" ana="Ncfpry" sn="1.1.x" oc="Group" dom="military"> patrulele</w>
  <w lemma="nu" ana="Qz" sn="1.x" oc="not" dom="factotum">nu</w>
  <w lemma="conta" ana="Vmii3p" sn="2.x" oc="SubjAssesAttr" dom="factotum"> contau</w>
  <c>.</c>
 </s>
</seg>
...
    </tu>
```

Figure 9. The final corpus encoding

4 WSD Evaluation

The BalkaNet version of the "1984" corpus is encoded as a sequence of uniquely identified *translation units*. For the evaluation purposes, we selected a set of frequent English words (123 nouns and 88 verbs) the meanings of which were also encoded in the Romanian wordnet. The selection considered only polysemous words (at least two senses per part of speech) since the POS-ambiguous words are irrelevant as this distinction is solved with high accuracy (more than 99%) by our tiered-tagger [21]. All the occurrences of the target words were disambiguated by three independent experts who negotiated the disagreements and thus created a gold-standard annotation for the evaluation of precision and recall of the WSD algorithm. The Table 4 summarizes the results.

Table 4. WSD precision, recall and F-measure

Precision	**Recall**	**F-measure**
78.21%	78.21%	78.21%

With the PWN senses identified (synset unique identifiers), sense labeling with either SUMO and/or IRST domains inventories is trivial, as described before, because the synset unique identifiers of PWN are already mapped (clustered) onto these two sense inventories. The Table 5 shows a great variation in terms of Precision, Recall and F-measure when different granularity sense inventories are considered for the WSD problem. Thus, it is important to make the right choice on the sense inventory to be used with respect to a given application. In case of a document classification problem, it is very likely that the IRST domain labels (or a similar granularity sense inventory) would suffice. The rationale is that IRST domains are directly derived from the Universal Decimal Classification as used by most libraries and librarians. The SUMO sense labeling will be definitely more useful in an ontology based intelligent system interacting through a natural language interface. Finally, the most refined sense inventory of PWN will be extremely useful in Natural Language Understanding Systems, which

would require a deep processing. Such a fine inventory would be highly beneficial in lexicographic and lexicological studies.

Table 5. Evaluation of the WSD in terms of three different sense inventories.

Sense Inventory	**Precision**	**Recall**	**F-measure**
PWN 115424 categories	78.21%	78.21%	78.21%
SUMO 2066 categories	85.08%	85.08%	85.08%
DOMAINS 163 categories	93.30%	93.30%	93.30%

Similar findings on sense granularity for the WSD task are discussed in [5] where for some coarser grained inventories even higher precisions are reported. However, we are not aware of better results in WSD exercises where the PWN sense inventory was used. The major explanation for this is that unlike the majority work in WSD that is based on monolingual environments, we use for the definition of sense contexts the cross-lingual translations of the occurrences of the target words. The way one word in context is translated into one or more other languages is a very accurate and highly discriminative knowledge source for the decision-making.

5 Conclusions

Word Alignment is a highly promising technology with real prospects of soon reaching full maturity and reliability as needed by commercial applications. Among them, one could mention multilingual computational lexicography and terminology, multilingual documents indexing and retrieval, open domain natural language question answering and obviously machine translation. We described another application, WSD, which is not an end in itself, but necessary at one level or another to accomplish most natural language processing tasks.

Neither YAWA nor MEBA needs an a priori bilingual dictionary,

as this will be automatically extracted by the TREQ or GIZA++. We made evaluation of the individual alignments in both experimental settings: without a startup bilingual lexicon and with an initial mid-sized bilingual lexicon. Surprisingly enough, we found that while the performance of YAWA increases a little bit (approx. 1% increase of the F-measure) MEBA is doing better without an additional lexicon. So, in the evaluation presented in the previous section MEBA uses only the training data vocabulary. The automatically extracted lexicons, could be almost 100% accurate (with a sufficiently high occurrence threshold) which is obviously a very good starting point in compiling bilingual dictionaries for language pairs where such electronic resources are not easily available.

YAWA is very sensitive to the quality of the bilingual lexicons it uses. We used automatically translation lexicons (with or without a seed lexicon), and the noise inherently present might have had a bad influence on YAWA's precision. Replacing the TREQ-generated bilingual lexicons with validated (reference bilingual lexicons) would further improve the overall performance of this aligner. Yet, this might be a harder to meet condition for some pairs of languages than using parallel corpora.

MEBA is more versatile as it does not require a-priori bilingual lexicons but, on the other hand, it is very sensitive to the values of the parameters which control its behaviour. Currently they are set according to the developers' intuition and after the analysis of the results from several trials. Since this activity is pretty time consuming (human analysis plus re-training might take a couple of hours) we plan to extend MEBA with a supervised learning module, which would automatically determine the "optimal" parameters (thresholds and weights) values.

The results in Table 5 show that although we used the same WSD algorithm on the same text, the performance scores (precision, recall, f-measure) significantly varied, with more than 15% difference between the best (DOMAINS) and the worst (PWN) f-measures. This is not surprising, but it shows that it is extremely difficult to objectively compare and rate WSD systems working with different sense inventories.

The potential drawback of this approach is that it relies on the ex-

istence of parallel data and at least two aligned wordnets that might not be available yet. Nevertheless, parallel resources are becoming increasingly available, in particular on the World Wide Web, and aligned wordnets are being produced for more and more languages (currently there are more than 40 ongoing wordnet projects for 37 languages). In the near future it should be possible to apply our and similar methods to large amounts of parallel data and a wide spectrum of languages.

Acknowledgements

The reported work is the result of several year intensive research at our institute. Many people deserve acknowledgements here, but special mentioning is due to Radu Ion, Alin Ceauşu, Dan Ştefănescu, Verginica Barbu-Mititelu and Elena Irimia, currently preparing their PhD theses on topics directly or closely related to those discussed in this paper.

References

[1] Dieterich, T., *G. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms,* Neural Computation, vol. 10, no. 7, pp. 1895–1923, 1998

[2] Fellbaum, Ch. (ed.) *WordNet: An Electronic Lexical Database,* MIT Press (1998).

[3] Tufiş, D. (ed): *Special Issue on BalkaNet.* Romanian Journal on Science and Technology of Information, Vol. 7 no. 3-4 (2004) 9–44.

[4] Ide, N., Veronis, J., *Introduction to the special issue on word sense disambiguation. The state of the art.* Computational Linguistics, Vol. 27, no. 3, (2001) 1–40.

[5] Stevenson, M., Wilks, Y., *The interaction of Knowledge Sources in Word Sense Disambiguation.* Computational Linguistics, Vol. 24, no. 1, (1998) 321–350.

[6] Vossen P. (ed.) *A Multilingual Database with Lexical Semantic Networks, Kluwer Academic Publishers,* Dordrecht, 1998

[7] Niles, I., and Pease, A., *Towards a Standard Upper Ontology.* In Proceedings of the 2^{nd} International Conference on Formal Ontology in Information Systems (FOIS-2001), Ogunquit, Maine, (2001) 17–19.

[8] Magnini B. Cavaglià G., *Integrating Subject Field Codes into WordNet.* In Proceedings of LREC2000, Athens, Greece (2000) 1413–1418.

[9] Moore, R. 2002. *Fast and Accurate Sentence Alignment of Bilingual Corpora in Machine Translation: From Research to Real Users.* In Proceedings of the 5th Conference of the Association for Machine Translation in the Americas, Tiburon, California), Springer-Verlag, Heidelberg, Germany: 135–244.

[10] Tufiş, D. *A cheap and fast way to build useful translation lexicons.* In Proceedings of the 19th International Conference on Computational Linguistics, COLING2002, Taipei, 25-30 August, 2002, pp. 1030–1036, ISBN 1-55860-894.

[11] Melamed, D. *Empirical Methods for Exploiting Parallel Texts.* Cambridge, MA: MIT Press, 2001

[12] Brown, P. F., Della Pietra, S.A., Della Pietra, V. J., Mercer, R. L.(1993) *The mathematics of statistical machine translation: Parameter estimation.* Computational Linguistics, 19(2) pp. 263–311

[13] Och, F., J., Ney, H., *Improved Statistical Alignment Models,* Proceedings of ACL2000, Hong Kong, China, 440–447, 2000.

[14] Och, F.J., Ney, H. *A Systematic Comparison of Various Statistical Alignment Models,* Computational Linguistics, 29(1), pp. 19–51, 2003

[15] Rada Mihalcea and Ted Pedersen, *An Evaluation Exercise for Word Alignment,* in Proceedings of the HLT/NAACL Workshop

on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond, Edmonton, Canada, May 2003.

[16] Martin, J., Mihalcea, R., Pedersen, T. *Word Alignment for Languages with Scarce Resources.* In Proceeding of the ACL2005 Workshop on "Building and Using Parallel Corpora: Data-driven Machine Translation and Beyond". June, 2005, *Ann Arbor, Michigan, June,* Association for Computational Linguistics, 65–74.

[17] Tufiş, D., Ion, R. Ceauşu, Al., Stefănescu, D.: *Combined Aligners.* In Proceeding of the ACL2005 Workshop on "Building and Using Parallel Corpora: Data-driven Machine Translation and Beyond". June, 2005, *Ann Arbor, Michigan, June,* Association for Computational Linguistics, pp. 107–110.

[18] Tufiş, D., Barbu, A., M., Ion, R. *A word-alignment system with limited language resources.* In Proceedings of the NAACL 2003 Workshop on Building and Using Parallel Texts; Romanian-English Shared Task, Edmonton (2003) 36–39.

[19] Fan, R., Chen, P.H, Lin, C.J. *Working set selection using the second order information for training SVM. Technical report 2005.,* Department of Computer Science, National Taiwan University (www.csie.ntu.edu.tw/ ~cjlin/papers/quadworkset.pdf).

[20] Tufiş, D., Ion, R. *Evaluating the word sense disambiguation accuracy with three different sense inventories.* In Proceedings of the Natural Language Understanding and Cognitive Systems Symposium, Miami, Florida, May 2005, pp. 118–127, ISBN 972-8865-23-6

[21] Tufiş, D., *Tiered Tagging and Combined Classifiers, in F. Jelinek, E. Nöth (eds) Text, Speech and Dialogue,* Lecture Notes in Artificial Intelligence, Vol. 1692. Springer-Verlag, Berlin Heidelberg New-York (1999) 28–33.

Dan Tufiş,

Received January 5, 2006

Institute for Artificial Intelligence,
13, "13 Septembrie", 050711, Bucharest 5, Romania
E–mail: *tufis@racai.ro*

Computer Science Journal of Moldova, vol.**14**, no.1(40), 2006

Linguistic Resources and Technologies for Romanian Language

Dan Cristea Corina Forăscu

Abstract

This paper revises notions related to Language Resources and Technologies (LRT), including a brief overview of some resources developed worldwide and with a special focus on Romanian language. It then describes a joined Romanian, Moldavian, English initiative aimed at developing electronically coded resources for Romanian language, tools for their maintenance and usage, as well as for the creation of applications based on these resources.

1 Introduction

As we begin the 21^{st} century, unhampered access to information technology is one of the foremost requirements for social development. Human Language Technology (HLT), sometimes called Language Engineering, concerned with providing information to the global community in natural, "human", language by supporting unrestricted access to text and speech media documents, is one of the most active research areas nowadays. To ensure that humans and machines have adequate access to resources expressed in natural language, including those on the Internet, technologies such as information retrieval and extraction, speech recognition and text-to-speech capabilities, machine translation, etc. must be developed for the widest possible variety of human languages and language families. Such a technological level is, at present, critical for languages less electronically visible, in order to ensure their appropriate exposure on the Web, especially on the Semantic Web, in a format and with processing capabilities able to give them quick and

complete integration with the new technological developments recently put on scene or which are being prepared worldwide.

The power of Language Engineering, as a solution to the communication requirements of the present times, mainly resides on the existence of sufficient resources in the languages to be treated. It has been estimated[1] that 98% of the HLT researchers who work with specific algorithms and technologies rely mainly on the resources created by the remaining 2% of the HLT community. Resources are used to make explicit, in symbolic form, the phonological, morpho-syntactic, lexical, semantic and structural information, of the language, expressed in both textual and speech form, as it is used in their most natural settings (novels, media, colloquial use, etc.). Apart from raw text or annotated corpora, in different languages, such resources may include statistical data in the form of language models, but also grammar rules, lexicons, dictionaries, terminological thesauri, wordnets, semantic networks, etc. Such resources have been developed on a large-scale for most Western European languages[2].

But the resources alone are not enough to accomplish the challenges expected from the domain of Human Language Technology. Next to them there must stay software tools capable to process the language at all levels and between levels. Section 2 resumes some of the processing machinery used recently in HLT.

Efforts to develop linguistic resources and processing tools for the Romanian language [16] are being pursued in a few centres of Romania (mainly in Bucharest, Iaşi and Cluj-Napoca), in the Republic of Moldova (Chişinău), and, sporadically, also outside this area,

[1] Rada Mihalcea's on-line presentation in the ConsILR meeting, Iaşi, November 2005; see http://consilr.info.uaic.ro/ro/resources/pre/Mihalcea/ConsilR%5B1%5D.Rada.2005.ppt

[2] Many useful linguistic resources and tools can be found at: LDC (Linguistic Data Consortium) - http://www.ldc.upenn.edu/, ELRA (European Language Resources Association) - http://www.elra.com, CLR (Consortium for Lexical Research) - http://crl.nmsu.edu/Tools/CLR/, ECI/MCI (European Corpus Initiative Multilingual Corpus I) – http://www.elsnet.org/resources/eciCorpus.html, MULTEXT (Multilingual Text Tools and Corpora) - http://www.lpl.univ-aix.fr/projects/multext/.

among the most active being those in Germany – Hamburg and Saarbrucken, in England – Sheffield and Wolverhampton, in Italy – Trento, in Canada – Ottawa and in USA – Dallas. In section 3, an overview of the achievements in the acquisition and development of language resources for the Romanian language is presented.

An initiative for the creation of Romanian resources and the development of dedicated tools has been recently agreed upon among the Faculty of Computer Science (FII) of the "Alexandru Ioan Cuza" University of Iaşi (UAIC)[3], the Academy of Sciences of Moldova (ASM)[4] and the University of Sheffield[5]. In a country such as the Republic of Moldova, known for the diversity of languages spoken by the predominant Moldavian population (Romanian) and different minorities (Russian, Ukrainian, Bulgarian, Turkish), the necessity for advanced language technology able to help the citizen to use her/his native language while also interacting smoothly with the official language administration and the multilingual society is remarkable. The main objectives, structure and expected results of the project, recently agreed for co-financing by INTAS[6] and ASM, called RoLTech, and which will be active between May 2006 and April 2008, are summarised in section 4.

2 Language Technology in the Information Society

The Information Society makes the best possible use of new Information and Communication Technologies (ICTs), which is capable to open to the citizen access to enormous amounts of information. Providing easy, natural, access to ICT services to people became a main worldwide objective. In 1998, UNESCO set up an Observatory on the Information Society, to keep the member states informed about the

[3] http://www.info.uaic.ro

[4] http://www.asm.md

[5] http://nlp.shef.ac.uk

[6] INTAS – the non-profit International Association for the promotion of cooperation with scientists from the New Independent States of the former Soviet Union - http://www.intas.be/

new ethical, legal and societal challenges brought by the Information Society. Since then, this observatory has been gradually updated, until it became an Internet-based gateway to online resources on these matters[7]. Although it has the air of being an invention of modern times, the Information Science and Technology (IST) paradigm, one of the seven major research areas in the classification of the EC, has its origins almost 70 years ago, when, in 1937, the American Society for Information Science and Technology (ASIS&T[8]) has been founded with the intention to help professionals in search for new and better theories, techniques, and technologies, and to improve their access to information. In Europe, the long-run declared objective of the EC with respect to science and technology is to overcome the United States and Japan, for the benefit of the society and its citizens. The EC is permanently boosting this challenge through its research policies and financing actions, and is guiding and supporting the revolution in IST through dedicated activities[9].

Language technologies can provide some of the necessary components for the present day Information Society. It can enrich the digital environment with the expressiveness of the human language that is usually lacking in human-machine interaction. Language technology applications and services can radically improve the efficiency and user-friendliness of the acquisition of information and communication tasks, covering business and leisure activities, government and education, services and life at home, through functions providing, minimally, information retrieval, interpretation and translation.

Language technologies are promoted in Europe at various levels, from the EC policies and dedicated programmes to the specific activities in universities and research centres. The 5th Framework Program (FP5[10]) had a Thematic Program on IST, with a specific action on Human Language Technology, to which has been attributed the largest

[7] http://portal.unesco.org/ci/en/ev.php-URL_ID=7277&URL_DO=DO_TOPIC&URL_SECTION=201.html

[8] http://www.asis.org/

[9] http://europa.eu.int/information_society/index_en.htm

[10] http://www.cordis.lu/fp5/

part of the budget (564 millions Euros) of the Key Action III of IST (Multimedia and Content Tools). EC programs like INCO and MLIS also included themes linked to this area. Currently, Human Language Technologies are addressed through the FP6 EC program[11], IST area ,,Knowledge and interface technologies" and FET objectives. The EC "e-Content" program[12] (2001-2004) was intended to stimulate the development, distribution and use of high-quality European digital content on the global networks. Its continuation, the e-Contentplus programme[13] aims to support the development of multi-lingual content for innovative, on-line services across the EU. In Europe there are many associations, universities and research centres with specific LRT activities[14].

Specific national programmes have also been created for the area of Language Technologies. In Romania, the National University Research Council (CNCSIS[15]) and the Managerial Agency for Scientific Research, Innovation and Technological Transfer Politehnica (AMCSIT-POLITEHNICA[16]) stimulate research in Language Technologies at (inter-) institutional and individual level.

The current approach in processing language nowadays is that language peculiarities should be separated from the algorithm, principle which assures both interchangeability of modules and reusability of language data in diverse settings. All modules designed on this principle receive an input file (usually containing a piece of free or annotated text) and a resource (a file able to configure the module according to the specificities of the language) and outputs a transformed (annotated) file or a graphical functionality for the benefit of an interacting user[17]. Designed on this principle, the processing modules used in HLT

[11] http://fp6.cordis.lu/

[12] http://www.cordis.lu/econtent/

[13] http://europa.eu.int/information_society/activities/econtentplus/

[14] For a list, see http://www.lt-world.org/

[15] http://www.cncsis.ro/

[16] http://www.amcsit.ro/

[17] Sometimes, for reasons of computational efficiency, the resources may be incorporated onto the processing module by a compiling process.

are seen to be language independent, their adaptability to a language or another being assured by the plugged-in resource files.

Following, we will make a rough inventory of language processing modules at different levels. The more basic, sub-syntactic, levels should include:

- tokeniser: a module capable to detect word boundaries, including compound words and abbreviations. Some largely used tokenisers are: the Penn Treebank tokenizer[18], GATE (General Architecture for Text Engineering)[19], and QTOKEN[20];

- morphological analyser: a module that returns lists of morphological features of words, each based only on the information communicated by the affixes, therefore taken isolated from the context. A morphological analyser will output all the morphological "interpretations" of an ambiguous word;

- part-of-speech (POS) tagger: taking as input a lexicon of words and a tagset, the POS tagger is a module capable to identify part of speeches of the words by using various methods: rule-based methods, statistical methods or Transformation Based Learning (TBL) methods [6]. Usually POS-taggers are based on previously collected statistics from a gold corpus (a corpus manually annotated by experts to POS data) – called a language model. The most successful POS-taggers have as core a dynamic programming algorithm, for instance – Viterbi [59]. It is proved that the tag set (minimally identifying the part-of-speech, but maximally a complex of morpho-syntactic features) can be optimised [48]. The Brill's tagger[21], the QTAG tagger[22], and the TnT tagger[23]

[18] http://www.cis.upenn.edu/∼treebank/tokenization.html
[19] http://gate.ac.uk/
[20] http://www.english.bham.ac.uk/staff/omason/software/qtoken.html
[21] http://www.cs.jhu.edu/∼brill
[22] http://www.english.bham.ac.uk/staff/omason/software/qtag.html
[23] http://www.coli.uni-saarland.de/∼thorsten/tnt/

are only few examples of POS taggers. The best results on tagging Romanian texts have been reported by Tufiş and Mason, [49] with a tagger based on QTAG;

- lemmatiser: a module that detects the words' lemmas (the canonical form of a lexeme). Lemmatisers, most often, work in combination with POS-taggers, since a lemma of an inflected word may not be unique and may depend on the context. If the context is not considered, the more elementary module is called a stemmer. Largely used lemmatisers are Ellogon[24] and TreeTagger[25]; useful stemmers are Heart Of Gold[26] and Snowball[27];

- chunk parser: a module that detects chunks of text, like noun phrases (NPs), verb phrases (VPs), or prepositional phrases (PPs). Chunks are non-overlapping spans of text, usually consisting of a head word (such as a noun) and the adjacent modifiers and function words (such as adjectives and determiners). The detection of chunks does not necessitate mechanisms more sophisticated than regular expressions [5, 2]. Well known tools including chunk parsers are: fnTBL, a customizable, portable and open-source machine-learning toolkit, primarily oriented towards NL-related tasks, currently trained for English and Swedish[28]; YamCha, a generic, customizable, and open-source text chunker oriented towards a lot of NLP tasks (POS tagging, Named Entity Recognition, base NP chunking, and Text Chunking)[29];

- segmenter: a module that detects sentence or clause boundaries. Most algorithms in this category use lists of key words (segments markers), which are words (expressions) manifesting delimiting

[24] http://www.ellogon.org

[25] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html

[26] http://heartofgold.dfki.de/

[27] http://www.snowball.tartarus.org/

[28] Freely available at http://nlp.cs.jhu.edu/∼rflorian/fntbl/

[29] Can be accessed at http://www2.chasen.org/∼taku/software/yamcha/

functions. Sometimes, the key words may also indicate the type of relation existing between the two segments it separates.

Above this level the syntactic processing should be considered:

- syntactic parser: a module that produces syntactic trees of the input sentence, either as constituency structures (for instance, Penn Treebank[30]) or as dependency structures. (an example is the Prague Dependency Treebank[31]).

On top of the syntactic level, processes addressing the semantics of natural language and the discourse level should be placed:

- word sense disambiguator (WSD): detects word senses according to a list of senses, as those in a dictionary or a wordnet[32];
- Named Entities Recogniser (NER): identifies expressions that can be classified according to a set of predefined categories, such as entities (organizations, persons, locations), temporal expressions (time, date), quantities (monetary values, percentages, numbers); Ellogon, Yamcha, GATE, Heart of Gold are some NER systems freely available (links above);
- semantic role detector: a module responsible for filling up semantic roles of main verbs (as described in FrameNet[33], for instance);
- discourse parser: a module that assembles discourse trees, usually applying decision criteria depicted from theories of discourse

[30] *Penn Treebank* was built at the University of Philadelphia – Pennsylvania, by Mitchell Marcus (http://www.cis.upenn.edu/∼treebank/home.html).

[31] *Prague Dependency Treebank* is currently being developing at the University of Prague, by Eva Hajičova and her team (http://quest.ms.mff.cuni.cz/pdt/).

[32] Among the projects related to wordnet development there are: the Princeton WordNet (http://wordnet.princeton.edu/), EuroWordNet (http://www.illc.uva.nl/EuroWordNet/), or Balkanet (http://www.ceid.upatras.gr/Balkanet/).

[33] *FrameNet* is a project initiated by prof. Charles Fillmore at the University of Berkley (http://portal.acm.org/citation.cfm?id=980860).

structure, e.g. Rhetorical Structure Theory [34]. Discourse parsing is based on segmentation at elementary discourse unit level and the use of a class of key words known to have discourse significance [36];

- summariser: a module capable to produce a summary from one or more documents. Summaries can be general, displaying most significant facts of a (collection of) document(s), or focussed, giving an insight of the reason for which a certain discourse entity is mentioned in the document(s). Summaries may be classified as extracts (created by reusing portions of the input text(s)) or abstracts (created by re-generating the extracted content) [33];

- anaphora resolver: a module capable to find the discourse entities that anchor referential expressions, such as pronouns, common and proper nouns, etc. Detection of text entities referred by pronouns, or nouns having a referential role (anaphors), is a vital process in many applications of text processing. For instance, in automatic translation, in order to correctly translate pronouns from a source language, in which pronouns have few forms, in a target language, which is richer in pronoun forms, it is of prime importance to know which entities those pronouns refer to. Only few translation systems are nowadays capable to correctly interpret a discourse that is longer than a single sentence, because they do not have means to recognise anaphoric relations. Summarization systems produce better outputs when they incorporate also anaphora resolution mechanisms. Other domains that make heavy use of anaphora resolution are: information retrieval, inter-document summarization and automatic question-answering.

A different type of processing is dealing with multilingual and parallel texts:

- language detection: the identification of the language of a span of text (mainly based on statistics drawn on the occurrence of letters or detection of words as belonging to certain vocabularies);

- sentence and word aligner: receives pieces of parallel texts and outputs their alignments at sentence and word level.

In the field of tools for processing language, of a significant importance is the GATE system [24], developed at the University of Sheffield – a modularised framework for the configuration of pipe-line architectures composed of language processing modules. The processing modules are supposed to be language independent (in the sense described above), therefore easily adaptable to Romanian by integration of adequate resources.

3 The language technology applied to Romanian language

Elaboration of lexical resources makes an important part of researches directed on Romanian language. Although not to the extent of other languages with greater electronic visibility, efforts have been invested by researchers in different places (Romania, Republic of Moldova, United States, United Kingdom, Germany, Italy, etc.) to develop Romanian linguistic resources such as corpora, dictionaries, wordnets and collections of linguistic data in both symbolic and statistical form (n-gram tables to configure language models, sets of grammar rules, name entity lists, etc.).

Research in HLT in Romania is being pursed in several centres, among which: in Bucharest, at the Research Institute for Artificial Intelligence of the Romanian Academy (AR-ICIA)[34] and the University of Bucharest; in Iaşi, at UAIC-FII and the Institute of Computer Science of the Romanian Academy – the Iaşi branch (AR-IIT)[35] ; in Cluj-Napoca, at the "Babeş-Bolyai" University, etc.

The research institutes in linguistics and philology, as those of the Romanian Academy, which, until very recently, employed only classical methods of study in the acquisition of linguistic dictionaries and

[34] http://www.racai.ro

[35] http://www.iit.tuiasi.ro/iit/index.php

thesauri, begun to show an ever-growing interest for digital techniques, mainly in using the computer for lexicographic tasks and the processing of linguistic material. For instance, [29, 30] report preparatory activities performed at the "Al. Philippide" Institute of Romanian Philology in Iaşi towards the computer-aided acquisition, in electronic form, of the Dictionary of Romanian Language (DLR), edited by the Romanian Academy. When this remarkable work, started in 1965 within the three academic institutes (the Institute of Linguistics ,,Iorgu Iordan – Al. Rosetti" – Bucharest, the Institute of Romanian Philology "A. Philippide" – Iaşi and the Institute of Linguistics ,,S. Puşcariu" – Cluj-Napoca), will be finished (expected for 2007), the new series of the DLR will contain 26 volumes, counting almost 12,000 pages, and including the letters D, E and L – Z[36]. Among the advantages of an electronic version of DLR, some obvious ones are: the possibility to add, correct, and modify any entry of the current version, to align word senses with similar entries of other resources (very useful for combined search), to exploit the large collection of examples associated with definitions of senses in order to acquire abilities to disambiguate, by statistical means, semantically ambiguous words in contexts (as required, for instance, in machine translation), to enlarge the database of texts/attestations used in the old series of the dictionary, and, finally, to print and publish easier, including on the Internet, the whole DLR or only instances of it, as imposed by different scientific or commercial needs. The cooperation between linguists and computer scientists at the "A. Philippide" Institute has made possible to develop a dedicated tool, DLRex [29], an instrument able to acquire, process and browse the electronic version of DLR.

In 2001, research groups from Iaşi, Bucureşti and Chişinău have founded the *Consortium for the Romanian Language: Resources & Tools*[37] – an initiative aiming to synergise the efforts of linguists and

[36]The rest of the letters form what is actually called the old series of the Romanian Language Dictionary, or the Dictionary of the Academy (DA) and has been published before 1949.

[37]The Consortium portal is at http://consilr.info.uaic.ro/, with Romanian and English versions

computer scientists who work on Romanian language, mainly by promoting to linguists the software tools for linguistic processing developed by computer scientists, and to computer science people – the resources created by linguists. Two fundamental aspects regarding the processing of a language are mainly taken into consideration in the activities of the Consortium: the elaboration of tools capable to process Romanian in conformity with the established international standards, and the creation and maintenance of proper linguistic knowledge (resources). At the last workshop dedicated to the activities of the Consortium, in November 2005[38], it was extremely encouraging that so many groups and individual researchers have presented their activities related to the development of resources and tools dedicated to the Romanian language. Their achievements will be briefly presented below.

At AR-ICIA, a large collection of corpora, mainly annotated, has been created. Among the manually validated corpora there are:

- *NAACL 2003* and George Orwell's novel 1984 are parallel English-Romanian corpora containing about 1.6 mil., respectively 250,000 tokens; the corpora are segmented, morpho-syntactically annotated and lemmatised. The 1984 corpus is word-aligned and annotated to word senses using the Princeton WordNet;
- Plato's *Republic* (250,000 tokens), *Evenimentul Zilei* (news, about 92,000 tokens), and *ROCO* (news, 7.1 mil. tokens) are parallel English-Romanian corpora, morpho-syntactically annotated.

At the same institute, using specially developed tools, the collection of automatically annotated parallel corpora contains:

- the Romanian FrameNet: 1,094 sentences from the original FrameNet 1.1. corpus (translated in Romanian at UAIC-FII), annotated morpho-syntactically and lemmatised;
- Timex: the Romanian translation (realised at UAIC-FII) of the TimeBank 1.1 corpus [42] – 186 news articles with 72,000 Roma-

[38] http://consilr.info.uaic.ro/en/index.php?showpage=030103

nian tokens; the corpus is lemmatised, morpho-syntactically and temporally annotated in English and partially in Romanian;

- RoSemCor: a parallel English-Italian-Romanian corpus[39], aligned to word-senses, including 12 articles from the SemCor[40] corpus. The alignment methodology will be used for another 80 articles from the same English corpus. The translations have been realized at UAIC-FII and the University of North Texas;

- Acquis Communautaire (about 12,000 Romanian documents; 6,256 parallel English-Romanian documents); the corpus is lematised, POS-tagged, sentence- and word-aligned.

The dictionaries/lexicons developed at ICIA include *WEB-DEX* – the explanatory Romanian dictionary, containing about 65,000 entries, XML encoded according to CONCEDE, *tbl.wordform.ro* – an ASCII file with cca. 546,000 occurrences of Romanian words, *Romanian paradigmatic morphology* – an unification-based description of the morphology with a lexicon of cca. 40,000 lemma, *RoWordNet* – the semantic network of Romanian, aligned to concept level with the Princeton WordNet, and containing about 30,000 synsets, from which about 20,000 were created in cooperation with UAIC-FII during the BalkaNet FP5 project [54, 20], *Romanian-French dictionary* – 16,710 entries XML encoded, and *EUROVOC* – a multilingual thesaurus containing bilingual dictionaries from 21 languages included in the Acquis Communautaire corpus.

The ICIA tools dedicated to HLT, will partially be available soon as integrated web services based on the WSDL, UDDI and SOAP protocols, and will include:

- EGLU – an integrated programming application, implemented in Common Lisp, based on unification of complex NLP systems; it

[39]For references, see http://multiwordnet.itc.it/english/home.php

[40]For the original corpus, see http://multisemcor.itc.it/semcor.php or references on Rada Mihalcea's pages at http://www.cs.unt.edu/∼rada/downloads.html.

includes a compiler of linguistic descriptions and modules for morphological analysis and generation, syntactical analysis – CKY, syntactic generation - Head driven, lexical or structural transfer for machine translation [47];

- DIC – a compiler of electronic dictionaries initially created for the automatic generation of the XML Concede encoding of DEX; with minimal modifications it can be used for compiling dictionaries with structures similar to those of DEX;

- TTAG (Tiered Tagset) – a system of automatic projection of an optimal tagset of morpho-lexical descriptors, implemented in Perl [48, 56];

- TT&CLAM (Tiered Tagging and Combined Language Models) - a system of morpho-lexical disambiguation on two levels, that uses combined language models; it includes a specialized editor [51];

- TTL – a Perl module that allows the text segmentation at sentence/word level, the lemmatization and morpho-syntactic annotation; the module is language independent and uses regular expressions and Markov models;

- WSDTool – a Perl application which annotates at sense level every word of an XCES parallel corpus; any parts from the corpus for which aligned wordnets exist can be annotated [32];

- TREQ – a Perl application that extracts translation equivalents dictionaries from parallel corpora [52];

- YAWA – a Perl lexical aligner, language independent for the modules that do not require specific alignments between the languages involved. It uses a parallel XCES corpus, morpho-syntactically annotated and lemmatized, and translation dictionaries obtained through TREQ [53];

- WN-Builder, WN-Correct – a set of programs, used during the BalkaNet project, dedicated to the development and correction of wordnets aligned with the Princeton WordNet [55];

- Ro-Hyphenator – a Romanian syllable splitter;

- DIAC – a system that automatically recovers the missing diacritics from Romanian texts [50];

- Multilingual thesauri aligner – a C# application used to align the English version of Eurovoc with the incomplete Romanian version;

- MTKit – a C# integrated application for the annotation/lexical alignment of XCES (Extended Corpus Encoding Standard)[41] files. It creates statistical translation models and, through a friendly graphic editor, helps to browse the lexical alignments and the properties of each constituent of the alignment (such as POS tags, lemma, chunks, word sense, and definition of the synset). The incomplete or incorrect alignments can be modified. Using a gold-standard, MTKit automatically calculates the alignment accuracy (precision, recall, F-measure) [57];

- Sentence Aligner – a C# application intended for statistical sentence alignment [7];

- LexPar – a Perl application which determines the structure of a given sentence as a graph of dependencies;

- XCESGen – a group of Perl modules that generates XCES format for parallel corpora;

- Google Screen Scraper – a library of instruments dedicated to automatic web search using Google capabilities.

[41] http://www.cs.vassar.edu/XCES/

Among the HLT modules developed at UAIC-FII, some being available on the ConsILR portal[42], there are:

- AnMorph – an environment for the development and updating of the paradigmatic morphological model of a non-agglutinative language (the basic inflected word model sees a word as composed of a root and an ending). Currently, the database of the application covers only partially the Romanian morphology. During the interaction with the system, a trained user works with a friendly interface to classify new words into already existent inflecting paradigms, or fill in, by examples, new paradigms. The program continuously compares the forms introduced by the user in the automatically drawn tables with those that can be generated from the stored paradigms and, if a matching is confirmed, generates the rest of the forms. When this happens, the user has only to check and validate the automatically filled-in parts of the tables. Apart from the developing/updating interface, the environment offers an editor for the dictionary and the collection of paradigms, a component enabling consistency checks of the data and a lemmatiser [12];

- Occurrence Finder – a tool aiming to identify occurrences of lexical sequences that are subject to restrictions of different kinds, in raw or annotated texts. This application could be of extreme interest to lexicographers, which build their dictionaries entries by looking for adequate examples in text corpora, but can be of help in any linguistic research activity that uses large collections of textual data. The system incorporates a specialized restrictions language – SXPath, which is based on the Xpath standard (the XML Path language). The incorporated search engine evaluates an SXPath expression in one single pass of the XML document, without having to load the whole document into memory. It goes through the document serially and keeps in memory only

[42] Applications available at http://consilr.info.uaic.ro/en/index.php?showpage=060101

On-line tools available at http://consilr.info.uaic.ro/en/index.php?showpage=0604

those parts that are relevant for the current work context. The engine returns the list of elements that observes the restrictions expressed by the SXPath expression [44];

- An environment for the processing of parallel XML annotated documents, including a program allowing the definition of annotation schemas (tag sets with corresponding attributes). A consistent set of schemas are placed in a hierarchy (lattice). Different annotations over the same text can be mixed on the same output document, or vice-versa, different partial annotations can be extracted from a complex annotation [18];

- RARE – a framework allowing the development and testing of anaphoric models [41, 19, 21]. Its architecture offers the user a build-in library of functions that can complement most of the foreseeable models for entity tracking in texts. It also allows a user to define its own strategies, specific to the application settings in which this functionality is a must. The engine recognises the first mentionings of characters/objects in a text and links their subsequent ones to these, outputting co-referential chains;

- Learning-based clause-level text splitter – is a tool that receives a raw text as input and produces an XML annotated version of it in which clause borders are marked. The architecture includes two modules: a learning module and the actual segmenter module. The learning module infers rules for the recognition of clause borders using a corpus of texts annotated adequately. Both modules make use of a list of markers, which is kept as an external resource of the tool. The segmenter can be used independently or integrated in any NLP application [38];

- Rule-based clause-level text splitter – a similar tool for detection of clause borders based on symbolic rules [43];

- Discourse Parser – a tool intended to reveal the (tree) structure of a text [39, 22]. In such a structure the inner nodes represent rhetorical relations and terminal nodes represent elementary text

spans, as discourse units. Once the rhetorical structure of a text is obtained, intelligent browsing over the semantic structure of the document can be performed: general document summaries, summaries focused on specific characters or objects (discourse entities) of interest, data mining in texts, tracking of characters over events and situations distributed in text, determination of temporal positioning of events described by the text as absolute intervals or points on the time axis or relative to one another, detection of relations among events and situations, detection of spatial positioning and/or spatial relations of discourse entities, etc. The discourse parser incorporates a number of natural language processing modules, including a part-of-speech tagger, a segmenter establishing bounds between discourse units, a module able to recognise noun phrases (shallow parser) and RARE (the AR-engine described above). The parser implements rules and heuristics, which combine information contributed by words or expressions, having a role in the identification of structure, with referential chains detected by RARE;

- Summarisation Tool – is a Java based application [40], with a web-interface, that receives an XML file annotated to discourse structure (no relations between the entities involved) and gives focused summaries using Veins Theory [14, 17, 23]. The summaries can be focused on discourse entities mentioned in the text;

- GLOSS – a visual instrument intended for corpus annotation according to its discourse structure and referentiality. The application allows simultaneous annotation of multiple documents [15, 13, 58];

- XML annotator – an interface helping a manual annotation activity over a text, resulting in an XML document. It allows the definition of tags and their attributes, insertion of links between tags and navigation along them [9];

- Tracker of temporal expressions in texts – a program that detects expressions denoting events, their instances and signals specific

links between events. The application is under development, the current version being capable to partially annotate temporal information in Romanian texts, using 3 out of the 7 tags proposed in the TimeML standard [45]. The application is trained on a manual annotated version of a part of the 1984 corpus [28];

- RPC (Romanian Page Categorization) – a system that marks the category of a webpage. The categorisation process is guided by a database containing a set of key words specific to each category, which is automatically generated from webpages and is used in the training phase, based on the words' frequencies [46];

- Language identifier – a program capable to detect the language of a span of written text based on tables of bigrams and trigrams (sequences of two and three letters);

- Word sense disambiguator using WordNet – an application [31] that disambiguates the words' senses in texts using a knowledge base and context information from a training corpus (Orwell's 1984, annotated to POS and syntax structure);

- XML converter – a Java application developed in the frame of the LT4eL FP6 project[43], intended to automatically transpose the html documents representing the learning objects from the 9 languages involved in LT4eL (Bulgarian, Czech, Dutch, English, German, Maltese, Polish, Portuguese, Romanian) to the agreed XML format, that will be further used in the next phases of the project;

[43]The LT4eL project (http://www.let.uu.nl/lt4el/) is an FP6 project that will run between 2005 and 2007 and will provide Language Technology based functionalities to support semi-automatic metadata generation, for the description of the learning objects, on the basis of a linguistic analysis of the content. Semantic knowledge will be integrated to enhance the management, distribution and retrieval of the learning material. Ontologies, key elements in the architecture of the Semantic Web initiative, will be used to structure the learning material within Learning Management Systems (LMSs), by means of descriptive metadata.

- Keyword extractor – a language-independent Java application implementing a tf-idf algorithm [35] built for the benefit of the LT4eL project, which takes as input a collection of UTF-8 encoded text documents and produces a list of key words.

Among the collection of linguistic resources and tools developed by the research group in HLT at AR-IIT, we indicate only few:

- 16 files (summing up 36,357 words) out of the 44 files (with 95,455 words) of the 1984 corpus marked for NPs, and 10 files (21,468 words) marked for referential expressions, using the PALinkA annotator[44]. The whole corpus was segmented using the SCD algorithm [26], into 15,014 clauses and 14,609 implicit VPs;

- A part of the Romanian version of Hemingway's novel "Farewell to arms" (20,148 words out of 61,262 words) morphologically annotated using the TexTag editor;

- A collection of parsers based on Java grammars for DEX and DLR [25].

At AR-IIT, another group is working on Speech Technologies. Recently, they have initiated the creation of a freely-available Electronic Archive of the Romanian Sounds[45]. The site of the archive represents a joint realisation of the Speech Technologies group of AR-IIT, UAIC-FII and the Technical University "Gh. Asachi" of Iaşi (the CERFS Excellence Centre). The Sounds Archive will be used to improve the algorithms for speech recognition and (non-concatenative) synthesis and is planned to host the following speech resources for Romanian:

- a database with recordings belonging to both professional and non-professional Romanian speakers, inhabitants of the Iaşi county, as well as pronunciations specific to other geographical areas.

[44] http://clg.wlv.ac.uk/projects/PALinkA/

[45] http://iit.iit.tuiasi.ro/romanian_spoken_language

Based on it, the Speech Processing Group at AR-IIT will organise statistical studies on the Romanian sound system and, more generally, on various aspects of the spoken Romanian language;

- a Romanian speech database with records taken from persons manifesting different speech pathologies;

- a collection of syllable and word recordings to be used by concatenative synthesizers and speech recognition systems;

- an electronic dictionary with Romanian pronunciations will be correlated with the Atlases of the Romanian language [1], developed as a joint work at AR-IIT and the Institute of Romanian Philology "A. Philippide", Iaşi[46]. A performing editor for dialectal texts, *EditTD*, has been created at AR-IIT and has been used for editing the Linguistic Atlas.

In the Republic of Moldova, the main centre of research in language technology is the NLP group at the Institute of Mathematics and Informatics of the Moldavian Academy of Science (ASM-IMI). This group has significant contributions to the development of resources and tools for processing Romanian language. Among them, the software package "Tools for linguistic applications" [3, 4] has been used in the implementation of the Romanian Spelling Checker RomSP [11]. This package, at the moment of its distribution, was one of the first major processing tools dedicated to the Romanian language, among those developed in the Republic of Moldova and Romania. The resources developed at ASM-IMI include:

- a database with linguistic information for Romanian at the word level. The database accepts queries formulated in SQL;

- an extensive collection of Romanian word forms (approximately 1,000,000);

[46]The volumes of the Atlas are published in the NALR/ALRR series – *The New Romanian Linguistic Atlas / The Atlases of Romanian on Regions.*

- a Romanian lexicon (70,000 lemma entries), with morphological, syntactic and syllable splitting information [10];

- a dictionary of synonyms;

- translation dictionaries for Romanian, Russian and English;

- a Romanian grammar, containing 866 grammatical rules and a set of 320 affixes, which have been used for the development of a morphological vocabulary of cca. 30,000 words.

All resources developed at ASM-IMI are paired with specialized tools for browsing and maintenance (editing, modifying and updating). Another set of tools at ASM-IMI are dedicated to the assisted learning of the Romanian morphology and syllable splitting.

Other research and acquisition of language thesauri are being performed at the Department of Informatics and Foreign Languages, belonging to the Faculty of Informatics and Microelectronics of the Technical University of Moldova, Chişinău. One of the most recent efforts towards the development of linguistic resources in this department is oriented towards the creation of a corpus of 885 decisions of the Justice Court (789,520 words). The texts were scanned, segmented, morphologically annotated, chunked (for NPs and VPs, using regular expressions), and the named entities were marked. A manual correction of the texts will be followed by a semantic annotation, by aligning it with a juridical ontology, specially developed in the same group, which includes 44 concepts with 140 slots and is coded in the RDFS format.

At the Academy of Economical Sciences in Chişinău, in a collaboration with UAIC-FII, pronunciation (wav file), images and video files, illustrating part of the word senses in a Romanian dictionary, have been added, resulting in a Multimedia Dictionary of Romanian [8]. The resulted Microsoft Access database is structured on 7 fields: word, word_with_accent, definition (text fields), as well as audio_word, audio_definition, image and video (OLE fields).

At the University of North Texas, USA (UNT), the following resources dedicated to Romanian language have been developed and are

available on the ConsILR portal and the UNT web-site[47]:

- ROCO – a Romanian news corpus with 400 million words, tokenized and POS-tagged (at AR-ICIA); for the Senseval evaluation the corpus was also sense-tagged;

- Sense-tagged resources – used as Romanian resources in the Senseval 2003 and ACL 2005 competitions, and accessible also as web-collections by means of the *Word Games* interface, developed together by UNT (Rada Mihalcea), USC Information Sciences Institute (Tim Chkloski), University of Ottawa (Vivi Năstase) and AR-ICIA (the group coordinated by Dan Tufiş);

- the Romanian-English parallel news corpus (850,000 words), containing a collection of the dailies *Evenimentul Zilei* and *Monitorul*, and aligned at word level;

- Romanian-English dictionary with 38,000 entries.

Another American group working on Romanian is located at the Department of Computer Science, University of Southern California (USC-DCS). Marcu and Munteanu [37] have presented and demoed, during an invited talk in Eurolan-2005[48], the first Romanian-English statistical translation system. Starting from 15k docs, containing 10M English words, and 170K docs, containing 85M Romanian words, mainly collected by students at UAIC-FII during a term project, they have filtered an accurate collection, contained 6.5M words in parallel texts, aligned at sentence level. This parallel corpus was used in the statistical machine translation system to learn probabilistic word/phrase-based translation rules and to produce from them translations from Romanian into English of a fairly good quality (Bleu score ranging from 20.9 for heterogeneous genre and 49 for the European legislation).

[47] http://www.cs.unt.edu/∼rada/downloads.html#romanian.

[48] Eurolan 2005 – the 7^{th} Biennial International Summer School on *The Multilingual Web: Resources, Technologies, and Prospects*, http://www.cs.ubbcluj.ro/eurolan2005/

The NLP group at the University of Hamburg (Germany) has recently developed G.E.R.L. – a German-English-Romanian lexicon[49], using the Parole/Simple model. Intended mainly for didactic usage, the corpus contains many technical terms and incorporates morphological, syntactic and semantic (synonymy, verbal thematic roles, collocations) annotations.

4 RoLTech – a project dedicated to the Romanian Language

RoLTech – Platform For Romanian Language Technology: Resources, Tools And Interfaces – is a project co-financed by INTAS and ASM, which will proceed for 24 months, between May 2006 and April 2008. Participants in the project are UAIC-FII – as coordinator, the NLP Laboratory of the University of Sheffield and ASM-IMI. The project aims to acquire electronic resources for Romanian language, to develop corresponding tools for their maintenance and use, and to create applications based on these resources.

The project has the following general objectives:

1. to gather and integrate on a dedicated portal the existent resources in electronic form for Romanian language (including dictionaries, thesauri, corpora – raw and annotated, as well as linguistic data) and to develop new ones;

2. to build a platform that will group tools dedicated to process Romanian language at morphological, syntactic and lexico-semantic levels, and that will support integration of these tools into complex applications;

3. to build interfaces able to offer to the citizen (including the native non-Romanian speaker) access to the resources in a friendly and interactive way (including access through the Web).

[49] http://nats-www.informatik.uni-hamburg.de/view/Main/GerLexicon

The intended Web-portal will store and give access to:

- reusable linguistic resources for language technology (including dictionaries, thesauri, corpora – annotated and raw, as well as symbolic and statistical linguistic data),

- language technology tools for Romanian (both open source and authored code),

- documentation related to Romanian language (documents, references to external resources or tools, to significant projects and scientific events dedicated to the domain of HLT, titles of books, collections of papers on Romanian language, significant scientific events, etc.).

A prototype version of the portal will be set-up at the beginning of the project and will be enhanced till the end of the project and, hopefully, maintained permanently after.

All the resources created during the project lifetime will be permanently integrated into the Web-portal. They should conform to formats that will make them re-usable for integration in different NLP applications dedicated to Romanian language. This is the reason why they are called in the project Romanian Reusable Resources for Language Technology (3RLT). Three types of applications will be developed in the project:

- the first type will target the non-native speakers of Romanian. An example is an adaptable e-learning system for Romanian morphology to be used by students, with interfaces and teaching materials in Romanian, English and Russian, with multimedia elements and with possibilities to extend to other languages. The main beneficiary of these applications will be the minority citizens of Moldova, mainly the Russian speaking population;

- the second type of applications will focus to ordinary Romanian speakers. Among them, one application will aim to enhance the

search output on collections of Romanian documents as intermediated by existing search engines, by exploiting the morphological variations of words and synonyms. Another one will offer an interactive Romanian spell-checking service over the Web;

- finally, a third category of applications will be dedicated to the expert users of Romanian language. One example is a support system for dictionary development, including advanced lexicographic operations as, for instance, abilities to use the context for detection of word occurrences in corpora, to support complex browsing among dictionaries of different types and multimedia presentations of linguistic material on Romanian. The package is addressed to experts working on the language technology, but which, until very recently, have used only classical pencil and paper methods to acquire linguistic data, sort and organise them onto dictionaries and other printed lexicographic sources.

A clear Project Management and Dissemination plan will foster the communication strategies among the members of the RoLTech consortium and will increase the transfer of knowledge and the distribution of the project results.

The portal will enable Web access to resources related to the Romanian language and their use in computer linguistics and human language applications, together with dictionaries or pointers to dictionaries, tools for language processing, interactive learning environments for Romanian, links to conferences, books, significant papers and other materials on Romanian computational linguistics and Romanian language technology, accessible through the Internet. The web portal will have a bilingual interface, in English and Romanian.

We are confident that the portal will contribute:

- to enhance the undergraduate, master and doctoral level research as well as the education at these levels in Computational Linguistics and Human Language Technology, in both Romania and Moldova, as is now taught in different universities, including the "Al.I.Cuza" University of Iaşi, the "Babeş-Bolyai" University of

Cluj-Napoca, the University of Bucharest, the Technical University of Chişinău, and others;

- to bring closer the collaboration between linguists and computer scientists that are working and are doing researches on the Romanian language, particularly by emphasising how computational methods can speed up the acquisition of linguistic data and enhance their quality, and by raising the awareness that modern linguistics is bound to using computational methods;
- to disseminate the research on Romanian CL and HLT beyond the geographical area where the language is mainly spoken, this way stimulating world wide collaboration on Romanian language;
- finally, it could become a site used by people wanting to learn Romanian or to enhance their knowledge on the language.

The learning system for Romanian language is intended for use by non-Romanian native students. It will have, for this purpose, a trilingual (English, Romanian, and Russian) interface, and will include teaching materials in all these languages. The environment will be able to adapt to the student's individual needs and perspectives, first, by enabling her/him to choose from among several options for teaching materials. Once the student is acquainted with the teaching material, the system then offers a choice of several activities (tasks, exercises, tests, and games) and lexical material. The student selects a topic, learning and testing activity, and the lexicon to be used that best suits her/his interests and learning style. As the student goes through the material, she/he can also scan the log of previous work and summary reports. Thus the functionally is that of an autonomous learning system, providing a mechanism for self-learning and self-assessment.

One of the particular features of Romanian is the richness of its inflexions. Romanian, belonging to the group of Romance languages, not only resembles its distinguished Romance sisters in the prodigious morphology of verbs, but inherits also from Latin the declinations of nouns and adjectives (feature lost by the other members of this family, but existing, for instance, in the Slav languages, from which is has

got also many influences). This is why it is very important for the user searching Romanian documents to have access to search engines which incorporate the ability of finding information based on morphological derivatives of the word. This feature can be manifested either at the level of the input, by allowing inflected word forms as search criteria, or at the level of the searched documents, by offering the retrieval of documents including variations of the word presented in the input. Optionally, synonyms of the input word could be used in search. The project includes also the creation of morphological interfaces for Web search engines and an adaptation of a Web-service for Romanian spelling checking.

The dictionary development support system is intended to help the interactive creation of dictionaries and lexicons, using the 3RLT. Mainly, it will be addressed to lexicographers supposed to be involved soon in the elaboration of DLRI, starting from the printed editions of DLR (as explained in section 3). Its development will be based on the existing version of DLRex [30], which is presently capable of editing and browsing functions adapted to lexicographic activities. It is supposed that at the end of the project, the activity for the elaboration of DLRI will already be initiated, based on a research plan [27] expected to be elaborated by the Romanian Academy in collaboration with UAIC-FII. But this interactive specialized working environment could equally be promoted among the Moldavian lexicographers working on Romanian.

The multimedia elements needed for a language learning system can include sounds (pronunciation of words), pictures (illustrations), and video clips (animated illustrations). Such elements, adequate for 3RLT, usually require some additional programming. They have a relatively big volume and necessitate binary representation. They can be kept in the same database as the linguistic information itself (so-called BLOBs), or the database can contain only pointers to them (URLs).

The proposed project goes in-line with these researches and elaborations and moves ahead the development of lexical resources on Romanian as well as the elaboration of tools to be integrated into applications or for direct consumption by the end-user. The absolute novelty

of the proposed project is its large-scale integration of resources and tools for Romanian language, with immediate applications intended to the society and the citizen. For the first time, in Romania and in Moldova, neighbouring countries which had, during the history, long periods of common co-habitation, modern technologies of language processing will be used for the benefit of the society – by preparing the technological integration of Romania into the EU, and the citizen – by helping Moldavian minorities to acquire the official language (same as in Romania).

5 Conclusions

Most of the knowledge currently generated in the Information Society is encoded in natural language, more specifically in the form of electronic written texts or speech data. Computational systems able to process and support this huge amount of knowledge have to use large scale and reliable language resources and tools. Creating, maintaining and disseminating the language resources and tools are challenging processes, with impact in many fields of the human activity and civilisation, such as science, culture, economy, and politics. When such resources and tools exist for a given language, they contribute strongly to the promotion of the national identity and the intercultural integration. When, on the contrary, they lack, the visibility of that language on electronic media is very poor, situation which could trigger extremely harmful effects against the people who speak that language and on the language itself. The modern times showed very clearly already that languages aggressively promoted on electronic media, as is the case of English, for instance, could influence important segments of an official language of a country, as is for instance the business jargon or the computers and communication jargon. In the past, when a language or dialect disappeared, the cause had to be looked for on specific social or political conditions (movement of population towards economically more developed zones, wars, etc.). Recently, to the traditional dangers, a new one seems to have appeared: the poor presence of the language on electronic media. One reason for this is the extreme attractiveness of

the Internet and other electronic communication media to the youngsters, therefore that segment of the population which will configure the shape of the language as will be it spoken by the next generation. It is difficult to foresee in detail the impact that a poor representation on the Internet could have on a language, but an alarm signal should be triggered already.

Languages are entities alive. They are transformed in the same rhythm in which the people that speak them are transformed by age and the renewal of generations. Sometimes, they are fragile and apparently minor changes influence them a lot, usually irreversible. If inappropriately cultivated, languages may even die[50] . With the Web technologies developing vividly, but especially with the Semantic Web, which is expected to open incredible computer-mediated content-based intelligent search and retrieval possibilities and which can be already perceived growing around the corner, languages are more and more dependent of their representation on the Web. It becomes a truism that the existence of language technologies for a language becomes nowadays a must in order for that language to "survive" in the Information Society.

The goal of this paper was to show the level of the research attained in the domain of Romanian LT, in order to foster its further development. It has begun by giving a short overview of the state-of-the-art of the language technologies in general, then it continued by presenting the main achievements on both resources and processing tools dedicated to Romanian, and finally it described a project that has just started. RoLTech, by its interdisciplinary nature (combining computer science and linguistics) and its international involvement (grouping researchers, native speakers of Romanian, belonging to Romania, Moldova, but also outside this area) is thought to make a significant step forward in the direction of the coordination of research actions addressing the computational aspects of the Romanian language from the modern perspective opened by the Information Society. Such a process,

[50]For information on endangered languages in Europe, for instance, see the UNESCO Red Book at http://www.helsinki.fi/%7Etasalmin/europe_report.html#Romansch

although having been initiated by several sporadic meetings that took place in the last three years in Romania as well as in Moldova, did not reached yet the maturity from which common actions could have arisen. The creation of a Web-portal intended to host language resources and processing tools, based on which applications will be developed, will help to achieve a much wanted common view on the future activities, to raise the quality of the research at European standards, and finally to offer the results to the citizens, as the final beneficiary.

6 Acknowledgements

In this paper we have used information presented on different occasions (as the EUROLAN-2005 Summer School and the November 2005 ConsILR meeting, organised by UAIC-FII), published or directly offered gently, on our request, by different people or groups. We are grateful to all of the following:

- our graduates, master students in Computational Linguistics and Ph.D. students at UAIC-FII Iaşi, past and present members of the Natural Language Processing Group. Special thanks are addressed to Bogdan Aldea, Cătălina Barbu, Roxana Bejan, Cosmin Bejan, Cristina Butnariu, Costel Coşman, Ovidiu Crăciun, Iustin Dornescu, Daniela Dudău, Gianina Dumitriu, Laur Gheţu, Vlad Horbovanu, Oana Hamza, Alex Hriţcu, Marinela Hriţcu, Maria Husarciuc, Ana Masalagiu, Livia Miron, Alex Moruz, Ciprian Niţă, Oana Postolache, Ionuţ Pistol, Georgiana Puşcaşu, Marius Răschip, Ioana Sandu, Violeta Sereţan, Valentin Tablan, Iulian Tănăsescu, Amalia Todiraşcu, Diana Trandabăţ, Daniel Ţorin, Cristian Ursu, and many others...
- the Natural Language Processing group of the AR-ICIA Bucharest, headed by prof. dr. Dan Tufiş, Correspondent Member of the Romanian Academy;
- the Speech Processing group of the AR-IIT Iaşi, headed by prof. dr. Horia Nicolai Teodorescu, Correspondent Member of the Romanian Academy;

- the Natural Language Processing team at AR-IIT Iaşi, headed by Nicolae Curteanu;
- the Natural Language Processing Group at ASM-IMI Chişinău, headed by dr. Svetlana Cojocaru and dr. Constantin Ciubotaru;
- dr. Cristina Florescu, dr. Gabriela Haja, and the DLRI/DLRex project teams from the "Al. Philippide" Institute of Romanian Philology[51] , Romanian Academy, Iaşi;
- Natalia Burciu and Natalia Elita from the Department of Informatics and Foreign Languages[52], belonging to the Faculty of Informatics and Microelectronics of the Technical University of Moldova, Chişinău;
- the NLP group at the University of North Texas, directed by dr. Rada Mihalcea;
- Valentin Tablan and Cristi Ursu, from the University of Sheffield, members of the GATE team;
- Dr. Daniel Marcu and Dragoş Ştefan Munteanu at the University of Southern California, authors of the Romanian-English statistical translation system;
- Constantin Orăsan, from the Research Group in Computational Linguistics, University of Wolverhampton[53], England, author of PALinKA;
- the Natural Language Systems Division[54], Department of Informatics, University of Hamburg, working on German-Romanian resources, coordinated by dr. Cristina Vertan.

The section 4 of this paper describes the INTAS project RoLTech no: 05-104-7633.

[51] http://www.iit.tuiasi.ro/philippide/index.html
[52] http://www.utm.md/en/3-1-3-1.html
[53] http://www.clg.wlv.ac.uk/
[54] http://nats-www.informatik.uni-hamburg.de/Main/WebHome

References

[1] V. Apopei, F. Rotaru, S. Bejinariu, F. Olariu. 2003. Electronic Linguistic Atlases. In *Proceedings of the International Conference on Information and Knowledge Engineering. IKE'03*, June 23-26, 2003, Las Vegas, Nevada, USA, Volume 2, pp. 628–633, CSREA Press 2003, ISBN 1-932415-08-4.

[2] S. Bird, E. Klein, and E. Loper. 2005. Chunk Parsing. Available at http://nltk.sourceforge.net/lite/doc/en/chunk.html.

[3] E. Boian, S. Cojocaru, L. Malahova. 2000. Instruments pour applications linguistiques. La terminologie en Roumanie et en République de Moldova, Hors série, No. 4.

[4] E. Boian, C. Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahova. 2003. Lexical resources for Romanian – a project overview. In: *Symposium on Intelligent Systems and Applications*, September 19–20, Iaşi, Romania. Eds.: H.N.Teodorescu, G.Gaindric, E.Sofron. Publisher: Tehnici si Tehnologii, Iaşi.

[5] T. Brants. 1999. Cascaded Markov Models. In *Proceedings of EACL 1999*.

[6] E. Brill. 1995. Transformation-based error-driven learning and natural language processing: A case study in part of speech tagging. *Computational Linguistics*, 1995.

[7] A. Ceauşu, D. Ştefănescu, D. Tufiş. 2006. Acquis Communautaire sentence alignment using Support Vector Machines. (to appear) in *Proceedings of LREC 2006*, Genoa, Italy.

[8] A. Chiorescu. 2005. The Explanatory Graphical Multimedia Dictionary. D.E.I. Multimedia (in Romanian: *Dicţionarul Explicativ Ilustrat Multimedia. D.E.I. Multimedia*). MSc. thesis, Faculty of Computer Science, "Al. I. Cuza" University of Iaşi.

[9] V. Ciubotariu. 2002. An XML annotation environment. Diploma thesis. Faculty of Computer Science, "Al.I.Cuza" University of Iaşi.

[10] S. Cojocaru. 1997. Romanian Lexicon: Tools, Implementation, Usage. In: Dan Tufiş & Poul Andersen (eds.). *Recent Advances in Romanian Language Technology*. Romanian Academy Publishing House, pp. 107–114.

[11] A. Colesnicov. 1995. The Romanian spelling checker ROMSP: the project overview. *Computer Science Journal of Moldova*, v. 3, Nr. 1(7), pp. 40–54.

[12] C. Coşman. 2002. The paradigmatic morphology of Romanian. Development and updating tool. (in Romanian: *Morfologia paradigmatică a limbii române. Mediu de dezvoltare / actualizare*). Dissertation thesis, Faculty of Computer Science, "Al. I. Cuza" University of Iaşi.

[13] O. Crăciun. 1998. GLOSS: Visual Instrument for discourse annotation (in Romanian: *GLOSS: Instrument vizual pentru adnotarea discursului*). Diploma thesis. Faculty of Computer Science. University Al. I. Cuza of Iaşi, Romania.

[14] D. Cristea, N. Ide, L. Romary. 1998. Veins Theory. An Approach to Global Cohesion and Coherence. In *Proceedings of Coling/ACL '98*, Montreal.

[15] D. Cristea, O. Crăciun, C. Ursu. 1998. GLOSS-A Visual Interactive Tools for Discourse Annotation. In *Proceedings of the Workshop on Annotation Tools, ESSLLI'98*, Saarbruecken.

[16] D. Cristea, D. Tufiş, 2002. "Romanian Linguistic Resources And Information Technologies Applied To The Romanian Language" (in Romanian). In Ichim O., F.T. Olariu (eds.) *The Identity Of The Romanian Language In The Globalisation Perspective* (in Romanian), Romanian Academy, the ,,A. Philippide" Institute for Romanian Philology, Trinitas Publishing House, Iaşi, pp. 211–234.

[17] D. Cristea. 2003. The relationship between discourse structure and referentiality in Veins Theory, in W. Menzel and C. Vertan (Eds.): *Natural Language Processing between Linguistic Inquiry and System Engineering*, "Al.I.Cuza" University Publishing House, Iaşi, pp. 9–22.

[18] D. Cristea, and C. Butnariu. 2004. Hierarchical XML representation for heavily annotated corpora, in *Proceedings of the LREC 2004 Workshop on XML-Based Richly Annotated Corpora*, Lisbon, Portugal.

[19] D. Cristea and O. Postolache. 2004. Designing Test-beds for General Anaphora Resolution, in *Proceedings of the Discourse Anaphora and Anaphor Resolution Colloquium – DAARC*, St. Miguel, Portugal.

[20] D. Cristea, C. Mihăilă, C. Forăscu, D. Trandabăţ, M. Husarciuc, G. Haja, O. Postolache. 2004. Mapping Princeton WordNet Synsets onto Romanian Wordnet Synsets. In *Romanian Journal on Information Science and Technology*, Dan Tufiş (ed.) Special Issue on BalkaNet, Romanian Academy, 7(1-2).

[21] D. Cristea, and O.D. Postolache. 2005. How to deal with wicked anaphora, in António Branco, Tony McEnery and Ruslan Mitkov (editors): *Anaphora Processing: Linguistic, Cognitive and Computational Modelling*, Benjamin Publishing Books.

[22] D. Cristea, O. Postolache, I. Pistol. (2005): Summarisation through Discourse Structure, In Alexander Gelbukh (Ed.): *Computational Linguistics and Intelligent Text Processing, 6th International Conference CICLing 2005*, Mexico City, Mexico, February 2005, Proceedings, Springer LNSC, vol. 3406, ISBN 3-540-24523-5, pp. 632–644.

[23] D. Cristea. (2005): Motivations and Implications of Veins Theory, in Bernadette Sharp (Ed.) *Natural Language Understanding and Cognitive Science, Proceedings of the 2nd International Workshop on Natural Language Understanding and Cognitive Science,*

NLUCS 2005, in conjunction with ICEIS 2005, Miami, U.S.A., May 2005, INSTICC Press, Portugal, ISBN 972-8865-23-6X, pp. 32–44.

[24] H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan. 2002. GATE: A Framework and Graphical Development Environment for Robust NLP Tools and Applications. *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics* (ACL'02). Philadelphia, July.

[25] N. Curteanu, E. Amihăesei. 2004. Grammar-based Java Parsers for DEX and DLR Romanian Dictionaries. In *Proceedings of the Third European Conference on Intelligent Systems and Technologies - ECIT'2004*, Iaşi, Romania.

[26] N. Curteanu, E. Zlăvog, C. Bolea. 2005. Sentence-Level and Discourse Segmentation / Parsing with SCD Linguistic Strategy, H.-N. Teodorescu et al. (Eds.), *Proceedings of the Intelligent Systems Conference*, Performantica Press, Iaşi, p. 153–168.

[27] C. Florescu. 2005. Proposals and suggestions for the Informatised and Unified Dictionary of Romanian (DLRI). (in Romanian: *Propuneri şi sugestii privind Dicţionarul limbii române informatizat şi unificat (DLRI)*). Oral presentation at the Workshop *Language Resources and Tools for Romanian Language Processing*, Iaşi, November 2005. Available at http://consilr.info.uaic.ro/.

[28] C. Forăscu, D. Solomon. 2004. Towards a Time Tagger for Romanian. In *Proceedings of the ESSLLI Student Session*, Nancy, France.

[29] G. Haja, E. Dănilă, C. Forăscu, B.M. Aldea. 2005. The Dictionary of Romanian Language (DLR) in Electronic Format. Acquisition Studies. (in Romanian: *Dicţionarul Limbii Române (DLR) în format electronic. Studii privind achiziţionarea.*) Alfa Publishing House, Iaşi.

[30] G. Haja, C. Forăscu, B.M. Aldea, E. Dănilă. 2006. The dictionary of Romanian Language: steps toward the electronic version. (to appear) in *Proceedings of Euralex 2006*, Torino, Italy.

[31] V. Horbovanu. 2002. Word sense disambiguation using WordNet. (in Romanian: *Dezambiguizarea sensurilor cuvintelor folosind WordNet*). Diploma thesis. Faculty of Computer Science. University Al. I. Cuza of Iaşi, Romania.

[32] R. Ion and D. Tufiş. 2004. Multilingual Word Sense Disambiguation Using Aligned Wordnets. In *Romanian Journal on Information Science and Technology*, Dan Tufiş (ed.) Special Issue on BalkaNet, Romanian Academy, vol. 7, no. 2-3, pp. 198–214, ISSN 1453-8245.

[33] I. Mani. 2001. Automatic Summarization. John Benjamins.

[34] W.C. Mann, and S.A. Thompson. 1987. Rhetorical Structure Theory: Toward a Functional Theory of Text Organization. In *Text 8(3)*.

[35] Ch. D. Manning and H. Schütze. 2002. Foundations of Statistical Natural Language Processing. The MIT Press.

[36] D. Marcu. 2000. The Theory and Practice of Discourse Parsing and Summarization. The MIT Press.

[37] D. Marcu and D.Ş. Munteanu (2005). State of the Art in Statistical-Based Machine Translation: A Romanian-English Experiment, invited talk, EUROLAN-2005, Cluj-Napoca, 25 July – 6 August.

[38] I. Pistol. 2003. Automatic discourse segmentation (in Romanian: *Segmentarea automată a discursului*). Diploma thesis. Faculty of Computer Science of the "Al. I. Cuza" University of Iaşi.

[39] I. Pistol. 2005. Automatic parsing of linguistic discourse (in Romanian: *Parsarea automată a discursului lingvistic*). MSc. thesis.

Faculty of Computer Science of the "Al. I. Cuza" University of Iaşi.

[40] O. Postolache. 2001. Automatic Summarisation (in Romanian: *Sumarizare automată*), Diploma thesis, Faculty of Computer Science, "Al.I.Cuza" University of Iaşi.

[41] O. Postolache. 2004. RARE – Robust Anaphora Resolution Engine. MSc thesis, Faculty of Computer Science, "Al.I.Cuza" University of Iaşi.

[42] J. Pustejovsky, P. Hanks, R. Saurí, A. See, R. Gaizauskas, A. Setzer, D. Radev, B. Sundheim, D. Day, L. Ferro M. Lazo. 2003. The TIMEBANK Corpus. In *Proceedings of Corpus Linguistics 2003*.

[43] G. Puşcaşu. 2003. Segmentation in elementary discourse units (in Romanian: *Segmentarea în unităţi de discurs elementare*). MSc. thesis, Faculty of Computer Science, "Al.I.Cuza" University of Iaşi.

[44] M. Răschip. 2003. Occurrence finder (in Romanian: *Ocurenţiator*). Diploma thesis, Faculty of Computer Science, "Al.I.Cuza" University of Iaşi.

[45] R. Saurí, J. Littman, B. Knippen, R. Gaizauskas, A. Setzer, J. Pustejovsky. 2005. TimeML Annotation Guidelines, Version 1.2.1, October 2005.

[46] S. Tamba. 2002. Web-page Categorisation. (in Romanian: *Categorizarea paginilor Web*). Diploma thesis, Faculty of Computer Science, "Al.I.Cuza" University of Iaşi.

[47] D. Tufiş. 1997. A Generalised Environment for Unification Based Natural Language Processing. In W. Teubert, R. Markincevicene (eds.) *Proceedings of the European Seminar on Language Resourses,* Kaunas.

[48] D. Tufiş. 1998. Tiered Tagging. In *International Journal on Information Science and Technology*, vol. 1, no. 2, Romanian Academy Publishing House, Bucharest.

[49] D. Tufiş and O. Mason. 1998. Tagging Romanian Texts: a Case Study for QTAG, a Language Independent Probabilistic Tagger. In *Proceedings of LREC 1998.*

[50] D. Tufiş, A. Chiţu. 1999. Automatic Insertion of Diacritics in Romanian Texts. In *Proceedings of the 5th International Workshop on Computational Lexicography COMPLEX*, Pecs, Ungaria.

[51] D. Tufiş. 2000. Using a Large Set of Eagles-compliant Morpho-Syntactic Descriptors as a Tagset for Probabilistic Tagging. In *Proceedings of the Second International Conference on Language Resources and Evaluation, LREC 2000*, Athens.

[52] D. Tufiş, A.M. Barbu. 2001. Automatic Construction of Translation Lexicons. In *Proceedings of the WSEAS and IEEE International Conference on Multimedia, Internet, Video Technologies*, ISBN: 960-8052-40-8, Malta.

[53] D. Tufiş, A.M. Barbu, R. Ion. 2003. TREQ-AL: A word-alignment system with limited language resources. In *Proceedings of the NAACL 2003 Workshop on Building and Using Parallel Texts; Romanian-English Shared Task*, Edmonton, Canada.

[54] D. Tufiş, E. Barbu, V. Barbu-Mititelu, R. Ion, L. Bozianu. 2004. The Romanian Wordnet. In *Romanian Journal on Information Science and Technology*, Dan Tufiş (ed.) Special Issue on BalkaNet, Romanian Academy, 7(1-2).

[55] D. Tufiş, E. Barbu. 2004. A Methodology and Associated Tools for Building Interlingual Wordnets. In *Proceedings of the 5th LREC Conference*, Lisabona, pp. 1067-1070.

[56] D. Tufiş, L. Dragomirescu. 2004. Tiered Tagging Revisited. In *Proceedings of the 4th LREC Conference*, Lisabona.

[57] D. Tufiş, A. Ceauşu, R. Ion, D. Ştefănescu. 2005. An integrated platform for high-accuracy word alignment. *JRC Enlargement and Integration Workshop: Exploiting parallel corpora in up to 20 languages*, Arona, Italy.

[58] C. Ursu. 1998. GLOSS: Visual Instrument for discourse annotation: validation and unification (in Romanian: *GLOSS: Instrument vizual pentru adnotarea discursului: validare şi unificare*). Diploma thesis. Faculty of Computer Science. University Al. I. Cuza of Iaşi, Romania.

[59] A.J. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Transactions on Information Theory 13(2):260–267. (The Viterbi decoding algorithm is described in section IV.)

D. Cristea[1,2], C. Forăscu[1,3],

Received May 2, 2006

[1]University "Al. I. Cuza" of Iaşi,
Faculty of Computer Science

[2]Institute for Computer Science,
Romanian Academy, Iaşi – Romania

[3]Institute for Artificial Intelligence,
Romanian Academy, Bucharest – Romania

e-mail: *dcristea@info.uaic.ro*,
corinfor@info.uaic.ro

Computer Science Journal of Moldova, vol.**14**, no.1(40), 2006

Local and Global Parsing with Functional (F)X-bar Theory and SCD Linguistic Strategy (I.)

Part I. FX-bar Schemes and Theory. Local and Global FX-bar Projections

Neculai Curteanu

Abstract

This paper surveys latest developments of SCD (Segmentation-Cohesion-Dependency) linguistic strategy, with its basic components: FX-bar theory with local and (two extensions to) global structures, the hierarchy graph of SCD marker classes, and improved versions of SCD algorithms for segmentation and parsing of local and global text structures. Briefly, **Part I** brings theoretical support (predicational feature and semantic diathesis) for handing down the predication from syntactic to lexical level, introduces the new local / global FX-bar schemes (graphs) for clause-level and discourse-level, the (global extension of) dependency graph for SCD marker classes, the problem of (direct and inverse) local FX-bar projection of the verbal group (verbal complex), and the FX-bar global projections, with the special case of sub-clausal discourse segments. **Part II** discusses the implications of the functional generativity concept for local and global markers, with a novel understanding on the taxonomy of text parsing algorithms, specifies the SCD marker classes, both at clause and discourse level, and presents (variants of) SCD local and global segmentation / parsing algorithms, along with their latest running results.

Notice. This is a paper in two parts, preserving a unitary numbering of the sections, and the unitary set and system of references along both parts.

1 Introduction: Basic Notions and Assumptions

This is a survey paper of the results, both theoretical and implementation aspects, concerning the latest form of *functional* X-bar (FX-bar) schemes (actually, graphs) and theory for *local* and *global* text structures, with a special accent on the problem of FX-bar (direct and inverse) projections of verbal group (verbal complex), and of the *SCD* (Segmentation-Cohesion-Dependency) linguistic theory and *segmentation / parsing strategy* for *local* (clause-level) and *global* (inter-clause and discourse) text structures. The paper is structured in two main parts, following the two main topics mentioned above.

In what follows, we shall try to specify, as much as possible, the basic notions we are working with. Within several papers and an evolution of the basic ideas along almost two decades [7], [9], [11], [16], [10], [17], the *SCD* (Segmentation-Cohesion-Dependency) linguistic strategy synthesized the following (more important) concepts and assumptions.

SCD considers four *major lexical categories* (and their functional projections within the FX-bar theory): *the Noun* (N) and *the Verb* (V) are the only lexical categories that have their own lexical (non-referential) meaning, and they are also saturated (representing their own semantic heads). Two other lexical categories play a central role in the syntactic organization of the *functional* X-bar (FX-bar) general schemes [11], [16].

The Adjective (Adj) has its own (auto-semantic) meaning but it is not a saturated lexical category, since it represents a modifier function to be applied to its intrinsic referentially nominal category, *i.e.* Adj is a modifier function that requires an N-type argument head. The pronominal adjective has a similar interpretation.

The Adverb (Adv) plays the role of V modifier, role similar to the one the Adj is playing for its N head. Often we denoted the modifier categories of Adj and Adv simply by A. It is important that this category is not confounded with the notation of A (Argument) positions (or A-bar, for non-argument positions), a common representation in classical linguistic theories. A special question is whether there exist

properly *predicational adverbs*, as adjectives do ("*predicational*" feature in the sense of [15], often called *deverbal* property). It seems (at least for Romanian) that such adverbs do not exist properly. The first and most feasible explanation would be that the two predicational features of the verb and adverb would interfere, being too 'close' to each other. This is not a completely satisfying justification since both predicational noun and its adjective pair may coexist naturally!

These *four major* lexical categories are important because they may be endowed with *two* essential lexical-semantics features of the *local* (*i.e. clause* level, which is also the *predicational* level) syntactic-semantic structures of language organization: *Tense* and *Predication* features.

The feature *TENSe* (*Time*) may receive at lexical (syntactic) or phrase (analytical) level the values *FINIte* or *NonFINite* as well as various analytical combined values of tense and aspect for the temporal forms of the *verbal complex* [28], [30], [2], [23], [18]. The *FINIte* value of the feature TENS, for each of the four major (lexical and) syntactic categories, is borne at the lexical level or inherited from the lexical level by the verbal complex (to what traditionally is called *predicate*). For the structure of Verbal Complex in [28], [2] etc. we shall continue to use the term "Verb Group" (abbreviated VG), in order to remain consistent with the notions, theoretical and computational approaches of the functional FX-bar theory and SCD linguistic strategy. Both correspond, in a great measure, to the concept of *verbal predicate* in classical grammar.

V is the only *chosen* category for which the feature *TENSe* may receive its value *FINIte*. The other major categories, N and A (Adj, Adv), receive the value *NonFINite*. These values of the feature *TENSe* involve the construction of the local syntactic structures: the Noun Group (NG), which is the classical NP with a single nominal head, VG (the Verbal Complex, already referred), and the finite and non-finite clauses.

1.1 Classical and Lexical Predications

The feature that we called *Predicationality*, borne at the *lexical* (even *lexicon*) level by the major lexical categories N, V, A, corresponds to what in the literature is called (more frequently, among other labels) the *deverbal* property, or *deverbality*, of these categories. For an extended survey and analysis of the notion and its syntactic-semantic consequences, see [15]. We avoid the term *deverbality* because its meaning is *not necessarily specific* to V*s* since this essential lexical feature is *equally shared* by N*s* and A*s*. Moreover, there are (classes of) verbs which do not bear this property, *e.g.* the *copulative* ones. The feature of *Predicationality* is assigned to those finite or non-finite V*s*, N*s* (often called *nominalizations*), and A*s*, whose meaning involves a *process* event or a *process* name. We abbreviated this feature as PRED(dication)F(eature), with two main values, PROC(cess) and STAT(e) (or EXIST).

The classical notion of *predication* is known to be the pair (*Subject*, *Predicate*), an essentially *syntactic concept* meant to support the finite clause (proposition) structure. The predicate, either synthetic or analytic, encloses both *process* verbs and *state* verbs (the latter case for the nominal predicate) indiscernibly, despite the fact that only process (predicational) verbs entail an argument-based syntactic distribution, corresponding to a proper *valence*. Furthermore, the feature of *predicationality* (or *deverbality*) is equally shared not only by process verbs but also by nominals N*s* and modifiers A*s* that are (in term of lexical semantics) siblings of the corresponding predicational verbs, these non-verbal categories having a *similar syntactic distribution* of arguments, with the same valence as their predicational, verbal counterparts.

Thus, the feature of *predicationality* (being a lexical semantics quality) is not necessarily related to the predicate (which is a syntactic construction): in the nominal predicate, the copulative verb is not a predicational one. The same goes for the auxiliaries incorporated within the VG (verbal group, or verbal complex) whose tense is based on compound syntactic constructions. This does not exclude, in the nominal predicate, that the predicative nominal (as semantic head of

the construction) bears the feature of predicationality. *E.g.*, the predicative nominals '*explanation*', '*marking*', '*receiving*' etc. (which are *predicational nouns*) in the nominal predicates of the clauses "*This is John's explanation* (*marking*, *receiving*, ...) *of the notion* ...".

These reasons support the idea of *handing down* the notion of *predication* from its classical, *syntactic* level to the *lexical*, word level of *representation* and *analysis*. The lexical semantics feature of *predicationality* (PREDF) has sometimes a contextual usefulness since the same word may, or may not, bear the feature PREDF, thus the process meaning depending on its contextual use. For instance, the noun "*building*" in languages like English, French, Romanian, may have both the meaning of a process, with [PREDF +] (or simply, PREDF), and the meaning of an object (in this case, the corresponding process result), with [PREDF –] (or STAT, or EXIST, or simply NPREDF values, see also [11], [12], [16]).

2 FX-bar Schemes for Local and Global Text Structures

2.1 Local (Clause-Level) Text Structures and FX-bar Projections

We pointed out within the SCD (Segmentation-Cohesion-Dependency) linguistic strategy that the natural language (NL) text is constructed from *local* and *global structures*. We consider *local structures* those structures that build a single finite-clause or a single (finite or non-finite) lexical predication (including both), in sum, finite or non-finite sub-clause and clause-level structures, while *global structures* represent inter-clausal or discourse level.

Thus a *local structure* is one of the following FX-bar structures: **(a)** single- (or multiple-) head *noun phrase*, together with its (their) FX-bar linguistic projection(s) (the single-headed noun phrase is called *noun group* NG in SCD); **(b)** single- (or multiple-) head *adjective phrase*, with its (their) FX-bar linguistic projection(s); **(c)** *finite verbal group* [7], [10], [11], known also under the label of *verbal complex* [28],

[29], [2], with its FX-bar projection elements (corresponding to what is also known to be the *verbal predicate*, either the *synthetic* or *analytic* one [20]; **(d)** *non-finite* VG, whose head is a non-finite V, bearing or not the predicational (deverbal) feature, and whose FX-bar projection is similar to that of the finite VG (verbal complex); **(e)** *finite clause*, viewed as the FX-bar projection of a finite VG; **(f)** *non-finite clause*, whose head is a *lexical* but *non-finite predicational* category (which can be a *predicational* but *non-finite* V, a *predicational* N, or a *predicational* Adj), together with its FX-bar projection.

Specif (or Spec) is also postulated in SCD to be a *functional* category bearing *quantificational features* at the lexical level (in particular, the negation at the X1 level), including (lexical or non-lexical) (in)definiteness, thus overlapping sometimes on the X1-marker functional features such as agreement.

The agreement (functional) relations are essential for what is called (*local*, *syntactic*) *cohesion* within SCD strategy: X0-Modif and X0-Specif agreement at the X1-level, Head-Subj and Compl-Pron$_{Emph}$ (Emphatic Pronoun) agreement at the X2-level etc. These kinds of (agreement, reference, and co-reference) *local cohesion* relations are responsible for a large category of *local dependencies*, including 'long-distance' dependencies. *Global cohesion* in the sense of [25], representing a chain of co-references for the same individual, is the discourse-level counterpart of a similar set of syntactic devices, but at the global level of text.

The FX-bar scheme for *local text structures* (including VG) is enclosed into the *line-bordered* (common) *part* of the figures 2.1. and 2.2. that present the *clause-level*, respectively *discourse-level global* FX-bar schemes.

2.2 Two Types of Global Text Structures

Global structures could be classified into (at least) two main categories: **(1)** There exist global structures built from *finite-clauses* or *lexical*

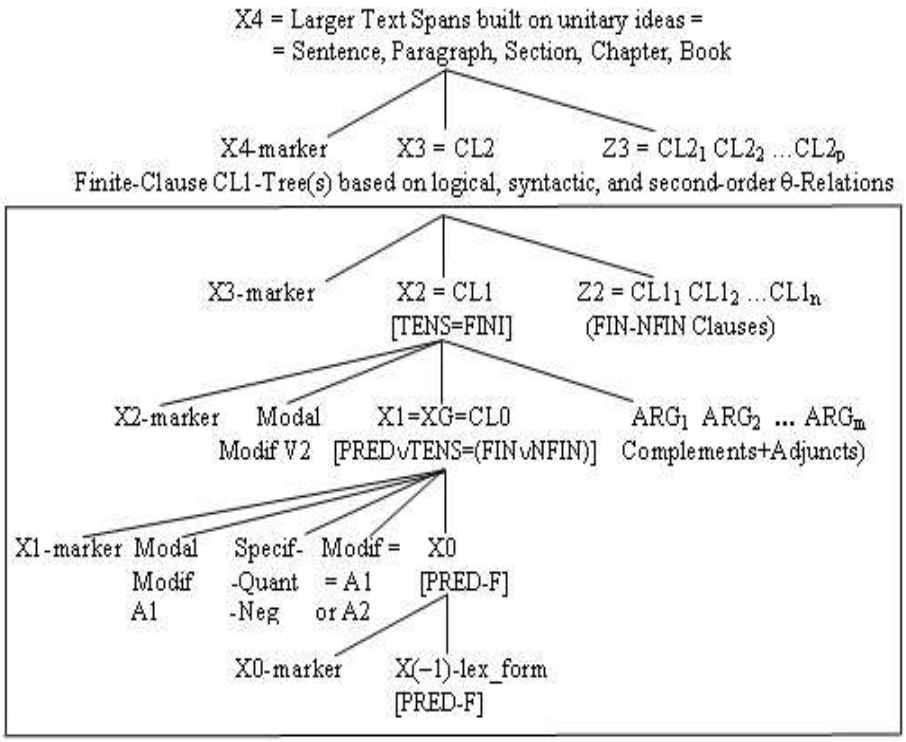


Figure 2.1. FX-bar scheme for clause-level local (and global) structures

predications (including both) using *logical* operators, *syntactic* operators (*e.g.* for the *relative clause*), and *second-order theta-relations* (*i.e. second-order predicational* relations, *e.g.* for the so-called subjective, predicative, direct-completive clauses etc.). **(2)** The usual *clause-based global text structures* are the sentence, paragraph, section, chapter etc.

The *clause-level global structures* correspond to the general FX-bar scheme whose elementary constructive element is the finite-clause (Fig. 2.1 above).

There exist *global structures* whose constructive bricks are not necessarily the finite-clause but the rhetorical *discourse-segment* of the RST discourse theory [24], [25], [26]. The FX-bar general scheme

extension to RST *discourse-segment global structures* is presented in Fig. 2.2.

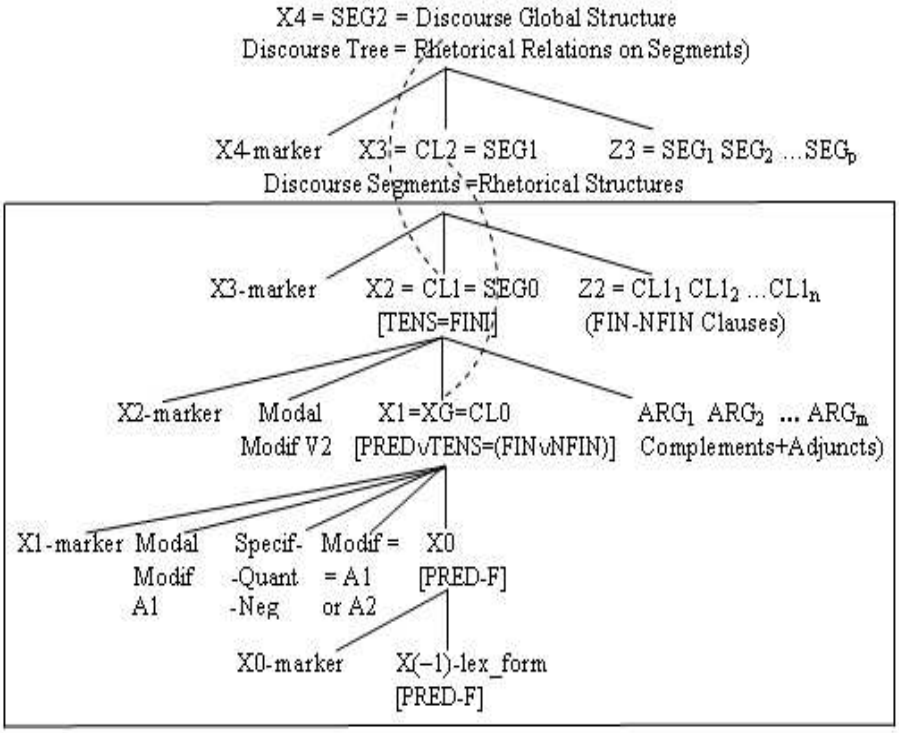


Figure 2.2. Discourse-level FX-bar (DFX-bar) Scheme

Remark. Dashed lines represent the special cases when a discourse segment is a proper subclause span and when a discourse segment splits a clause.

In general, a RST discourse segment is comprised of one or several finite-clauses. Actually, there exist an intricate relationship between the RST discourse segment and the finite clause, explored in [16]. Briefly, we have underlined that there exist sub-clausal discourse

segments (*e.g.* a discourse segment constituted from a single NG), or that a discourse marker may split up a (finite) clause into text spans that belong to distinct discourse segments.

Significant elements involved by the *new linguistic projections* incorporate the discursive functionality within the currently proposed DFX-bar scheme (Fig. 2.2), while the categories and structures specified by the projection principles in the 'old', *local*, *clause-level* FX-bar scheme and theory remain the same (the bordered part in the figures 2.1 and 2.2).

2.3 FX-bar (Classes of) Markers and Their Graph-Graph Hierarchy

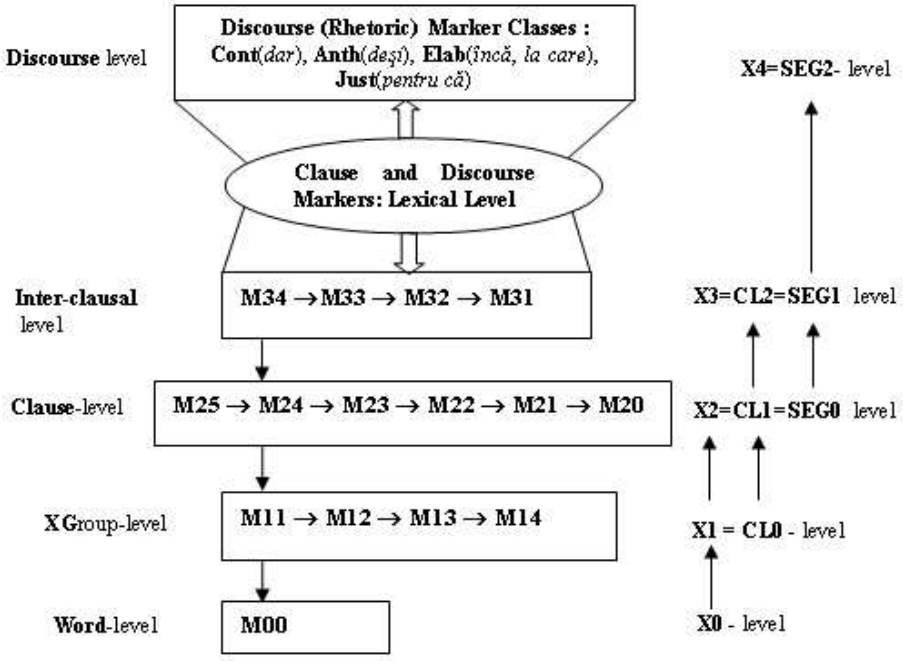


Figure 2.3. The hierarchy graph of SCD marker classes for local and global structures ([17])

One may naturally continue with relational (actually, functional) lexical (overt) categories, *e.g.* clause-level and discourse collocations (cue phrases, connectors, etc. called *clause* and *discourse markers*), but also lexically empty (covert) functional categories, such as T(ense) (or INFL) in [3], [4], [5], or the intrinsic presence of *predicational* (actually, *predicationality*) *feature* ascribed to a lexical category (and inherited by the XG phrase where that category is embedded in) [15] etc.

The device used to attain this goal is what we called *classes of functional markers*, together with their corresponding *hierarchies* [8], [9], [35], [17]. The dependency graph of the hierarchy of SCD marker classes (Xn-marker in Fig. 2.3) is represented in [14], [17]. The *explicit description* of the marker classes that are used within the FX-bar schemes in Fig. 2.1 and 2.2, and in the dependency graph in Fig. 2.3 is exposed in Section 7, Part II of the paper.

3 Local FX-bar Projections

3.1 Verbal Group Kernel and the Predicational Feature

The verbal group (VG), as XG structure in the BAR = 1 projection level of the FX-bar scheme, contains a *semantic head* verb, *around* which one can find pronouns (only in unaccentuated forms, *i.e.* clitics), special adverbs, auxiliaries, modal verbs (or adverbs), negation. VG is also better known under the label of *verbal complex* (see [28], [29], [2], [23]), and constitutes what is traditionally called *verbal predicate* for the classical clause (proposition). The VG Kernel (VGK) was initially introduced in [17, p.175] (under the name of *default verbal kernel*), and represents a basic substructure in the VG parsing. The *typical difference* between VG and VGK is that VGK is missing the *proper adverb* of VG (that may syntactically commute with VGK to accomplish the VG).

Examples of VGK*s* [17] (VGK is represented in parantheses, included in VG; unaccentuated pronouns (clitics) in VGK are in *italics*):

" nu că (nu *mi-l* va mai şi plăti) greu; (nu-*i* cunoşteam); (*li se* cereau) ; (*îşi* mai recăpătase) ; (Ai consultat) ; (ar fi simţit) ; (*i se* aşternea) ; (să *se* întâmple) ; (nu *se* putea abţine); (n-*o* putea lua); (Nu *i*-ar fi trecut); (să poată afla); (să *te* intimideze); (să *vă* văd lucrând) ".

VG may be seen as the shell of VGK, while the contents of VGK may be interpreted as the *clause-shadow* (of the regular clause) that projects itself onto the clause, as well as representing the projection(s) of the lexical-semantic head bearing the *predicationality feature* (*e.g.* [15]), using *diathesis transformations* and *semantic diathesis functions* associated with semantic restrictions on predication arguments (see [29], [28], [2], [23]).

A rightful observation in [2] is that VG provides both an *outside* ($\mathbf{nu}_1$) *negation* and an *inside* ($\mathbf{nu}_2$) *negation* (*e.g.* "$\boldsymbol{nu}_1$ *să* $\boldsymbol{nu}_2$ *te duci*"), which can be interpreted as outside (VG) and inside (VGK) *quantifiers*. Similarly, there exist as VG *modifiers* the (VGK) *special, inside adverbs* ("cam", "mai", "prea", "şi", "tot"), and the proper, VG *outside adverbs* ("nu_1 *să* nu_2 *te* $\boldsymbol{tot}_2$ *duci* $\boldsymbol{imediat}_1$"). The structure of VGK as the "inside" of VG, with a *syntactic head* (the auxiliary bearing the number and person, when lexically present) and a *semantic one* (the predicational verb), with clitics 'inside' (and arguments 'outside') VGK playing an essential role in the development of the *lexical predication* should be further explored, both in linguistic theory and parsing.

In a *verbal group* (VG), the "positive" feature values such as PROC and FINI are *inherited* from the tensed V head by the whole VG phrase, or may be *cumulatively acquired* through morpho-syntactic FX-bar projection.

Somehow similarly (preserving proportions) to the A. Joshi's well-known *tree adjoining grammar* (TAG) and *lexicalized* TAG (LTAG) [27], SCD strategy may also be seen as a *theory* of (D)FX-bar *scheme* (thus *tree*) *checking and adjoining*. Since in LTAG one considers the *initial trees* to be of the form 'functor-arguments', thus one begins in phrase generation with a clause shell, our VGK, whose structure is a *clause-shadow*, may constitute a substantiation argument for *initial trees* in lexicalized TAGs.

In [15] we discussed a suitable taxonomy for classical predications, involving the classical predicates (VG or verbal complex), based on the lexical property of predicationality PREDF, in agreement also with the extensional / intensional logical representations of these structures.

A typical example of the SCD predicational taxonomy is given by the two main categories of common nouns: **(i)** non-*predicational nouns*, corresponding to *existential-type*, *object-denoting*, *non-event individuals*, whose predicational feature PREDF value is EXIST (*e.g.* [Eng: *student, table*; Rom: *elev-student, masă*]), and whose functional representation in *extensional logic* is done by predicates depending on a single, *extensional variable*: *student*(X), *table*(X) etc.

Our interest is however in **(ii)** *predicational nouns* (often called *nominalizations*), whose predicational PREDF feature value is PROC, *e.g.* [Eng: *meeting, envy, marking,* etc.; Rom: *întâlnire*, *invidie*, *marcare*, etc.], whose functional representations depend on several *intensional variables*, *e.g.* *întâlnire*(x, y, ...), *invidie*(x, y, ...), *marcare*(x, y), *donaţie*(x, y, z) etc. Proper nouns and/or personifications are encoded either as constants or variables of extensional nature on which the above extensional / intensional predicates are applied. Other examples: the common nouns *car* and *man* are non-predicational individuals, represented extensionally as $car(X)$ and $man(X)$, the adjective *red* is a basic, also *extensional* predicate $red(X)$, while *leaving* is a *predicational* (process-event) *nominal* (also called *nominalization*) which is represented as an intensional (unsaturated) predicate *leaving*(x, y), with x and y as *intensional* arguments.

[Eng: *boy, pencil*; Rom: *băiat, pix*] PREDF = EXIST, and TENS=NFIN;

[Eng: *attempt, showing, proved*; Rom: *încercare, arătând, demonstrat*] PREDF = PROC, and TENS := NFIN;

[Eng: *are*; Rom: *sunt*] PREDF = EXIST, and TENS := FINI;

[Eng: *gives*; Rom: *dă*] PREDF = PROC, and TENS := FINI.

The *predicational nouns* are typical non-verbal categories whose distributional behaviour is perfectly similar to their verbal counterparts included in VG*s*.

3.2 From Syntactic to Lexical Predication

Without coming into details (see [18]), the classical predication pair (Subject, Predicate) can be viewed as just *one of the facets* of the VG (verbal complex) whose semantic head bears PREDF, the other ones, equally righted as "classical predications", being instantiated by the predicational verb (lemmatized form), endowed with clitic(s) as affixed inflexion(s), which are obligatory present when their valence-based arguments are of personalized semantic nature and optionally present otherwise, doubled or not by the corresponding valence-commanded arguments. Thus, the classical predication pair corresponds to the subject theta-role of "actor" or "actant", while the other "classical" predications associate, valence-driven, the theta-roles of "patient" and/or "receiver" and/or "addressee" to semantic arguments (but not adjuncts!). All these are commanded (or not) by the presence (or absence), at the *lexical level*, of the PREDF feature assigned to the semantic head in VG (verbal complex).

Thus, in a *first move*, the classical predication pair (Subject, Predicate) should be reduced to the pair (Subject, PREDF_verb) corresponding to the *theta*-role of "actor" or "actant" in the valence-driven SUBCAT vector (with 1 to 3) semantic arguments. It is important to specify that there exist normally at least two SUBCAT lists: $\text{SUBCAT}_{oblic\_order}$, containing the syntactic arguments of the PREDF_verb, in the order of increasing obliqueness, and $\text{SUBCAT}_{theta\_order}$, enclosing the arguments in the *theta*-order (or *systemic* order) for the valence-based arguments of PREDF_verb. Usually, (only) *for the active voice* and a normal semantics of predicationality, these arguments should coincide.

In a *second move* the following similar "classical" predications (see Figure 3.2) are added to this predication, equally righted in the *theta*-semantics.

These are the new 'traditional' predications, with their real engine, *viz.* the predicational feature PREDF, installed on the verb head of the verbal group VG (verbal complex). Similarly, non-finite forms of PREDF verbs may be associated to those N*s* (called nominalizations)

and/or As that bear the feature PREDF.

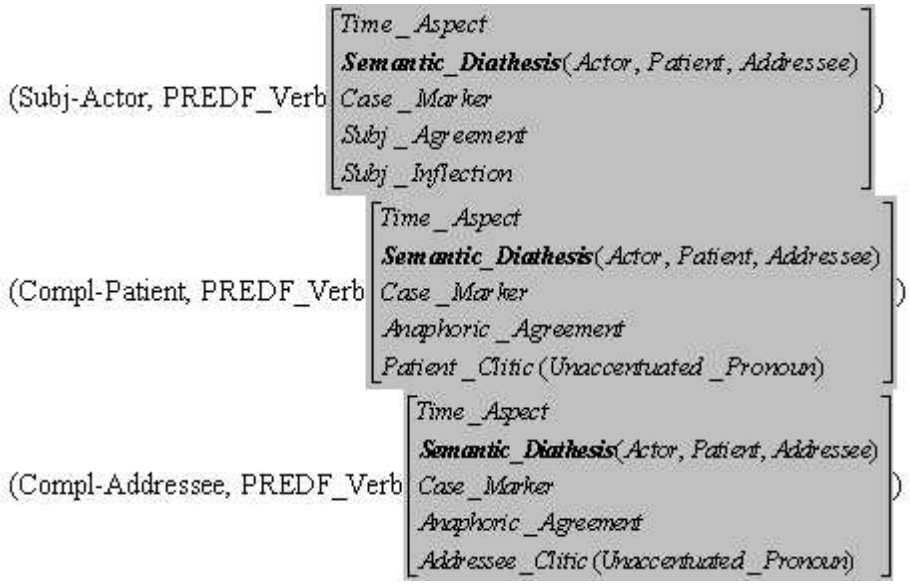


Figure 3.2. All the extended, valence-based 'classical' predications

In the 'classical' predications above, clitics may lack when the semantic arguments are of *non-person* or *non-animate* nature but are lexically present. This does not change the 'equivalence' of these newly devised valence-based predications. Such an interpretation of the VG structure has consequences in establishing the FX-bar (direct and inverse) VG projections (see the outlined solutions considered in the subsection 3.1 devoted to the problem of VG local structure and its FX-bar projections).

The problem of 'classical' *predication*(s) in HPSG [1], or the problem of the *special* role of the *subject* in the SUBCAT list of HPSG theory [34; Chap.9] are solved in the linguistic feature structures in Fig.3.2. above as follows: the feature *Semantic_Diathesis*(*Actor*, *Patient*, *Addressee*) is not an elementary (atomic) feature value but a *function*,

whose input value is the VG shallow, *syntactic diathesis*, represented by the above mentioned SUBCAT$_{oblic_order}$, while the output value of the function is the VG *semantic diathesis*, *viz.* SUBCAT$_{theta_order}$ list. This solution forces the subject-*actor* and the subject-*least_oblique_element* (or grammatical subject) to take each one its own right place, in the right (possibly distinct) ordering.

In the parsing / generation processes, the *input value* of the function *Semantic_Diathesis* is represented by the tense and syntactic diathesis resulted from the VG shallow parsing. The *output value* of *Semantic_Diathesis* function is obtained from the lexicon, where the head verb (*predication*) meaning is represented by specific standard lists of semantic arguments corresponding to the valence of that specific predicational category, and the syntactic diathesis is transformed into a certain particular list of semantic arguments corresponding to the tense, diathesis, and predicational meaning of that (verb) category. [18] describes in detail the mechanism of *diathesis transformation* and *semantic diathesis* functions, defined to make operational the effective resolution of VGK direct and inverse FX-bar projections (see §3.3).

In Fig. 2.1.-2.2. of the FX-bar scheme for *local structures*, the local (single-event) levels X0-X1-X2 express the clause predication depending on basic, lexical categories, while the levels CL0-CL1-CL2 express logical or (second-order) predicational relations on simple clauses. The two global FX-bar schemes work in a (top-down and bottom-up) recursive manner, both in the analysis and generation tasks of the parser, in close relationship with SCD linguistic strategy, its marker classes and hierarchy, and its meta-algorithms of analysis-generation being exposed in [9], [10].

3.3 Direct and Inverse FX-bar Projections of VGK

In [18] we introduced *diathesis transformations* (*dt*) and *semantic diathesis* (*sd*) *functions* as useful tools in describing the lexical predication metamorphosis from syntactic (shallow) diathesis to semantic diathesis as a top-down and bottom-up movement, from text to lexicon and backwards. This process may also be understood as direct and

inverse FX-bar projection procedures of VG (VGK) towards its (predicational) semantic head and to the clause, derived from the diathesis analysis (described as in [26]), stated as solutions to the following VG (VGK) FX-bar projection problems:

FX-bar(VG): The problem of direct FX-bar projection of VG: To show how the *clause-shadow* information (see above) incorporated into VG is (directly) FX-bar projected into a (finite or non-finite) regular clause.

FX-bar^{-1}(VGK): The problem of inverse FX-bar projection of VGK: To obtain an improved linguistic mechanism by which a predicational category (from the lexicon) is FX-bar projected on VG (VGK). This means to establish the FX-bar *inverse projection* FXbar^{-1}(VGK) for VGK (or VG), *i.e.* the morphologic-phonologic-syntactic-semantic restrictions on the (predicational) semantic head of VGK that are necessary (and sufficient) to retrieve the VG (VGK) *local* structure through (direct) FX-bar projection of its semantic head.

The inverse FX-bar projection associates to VGK a number of (virtual) semantic heads, corresponding to the meaning(s) of the lexical head entry, each semantic head observing the set of *diathesis transformations* (*dt*) and *semantic diathesis* (*sd*) *functions* and *values*, along with phonologic, lexical, morphologic, syntactic and semantic restrictions at lexical level on arguments, clitics, doubling etc.

This is the starting point in the process of *generation task*, when the first requirement is to generate one or several adequate VGs, satisfying the text *planning* restrictions. For clause analysis / generation, the parsed VG (as *clause-shadow*) or the obtained VG(s) is FX-bar projected into one (or more) finite or non-finite clause(s), with its (their) arguments, constructed lexically from diathesis computations and linguistic restrictions.

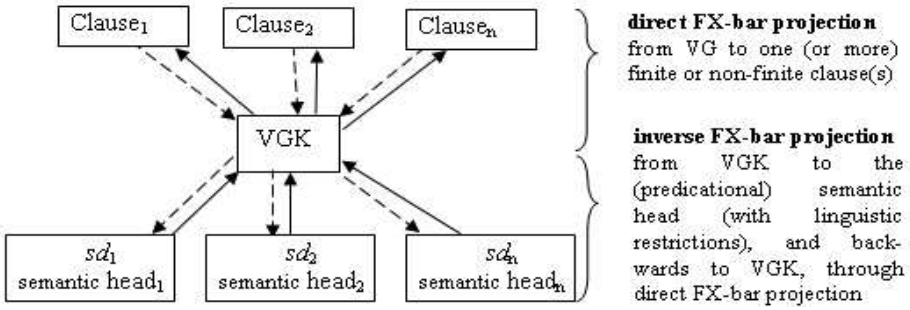


Figure 3.3. FX-bar projections of VGK, from text to lexicon and backwards

4 Global FX-bar Projections

4.1 Direct and Inverse Global FX-bar Projection

It is a common fact in the classical grammar to expand the thematic arguments (*theta*-roles) from inside the clause to the inter-clausal relations of the same type, using such labels as *subjective*, *predicative*, *direct-completive* etc. *clauses*. Such a clause tree, whose *inter-clausal* relations are based on *logical-type* operators (conjunction-disjunction, implication, conditional, concession, consecution-purpose, correlated operators, such as *if-then-else* etc.), (purely) syntactic-type relations (such as the *relative clause*), and *second-order theta-semantics* relations (as the above mentioned *theta*-role clauses) could be considered the *global linguistic projection* of the *saturated matrix (root) clause* of the *clause-level tree*. Other weaker (syntax or grammatical-oriented) semantics, together with node operations on the clause-tree may be taken into account.

A *similar problem* can be stated for the discourse segments, in particular for the *discourse tree* evolved from the RST inter-segment *rhetorical relations* [24]. Questions of theoretical and practical (computational) importance: which is the discourse projection nucleus for a resulted RST discourse tree, and what is the relationship between the corresponding clause and discourse segment trees?

In terms of discourse tree, we may state the following conjecture: a *text could be seen as* the "global" *projection of its discourse tree* (or of certain subtrees of the discourse tree). It is a kind of "summarization" of the text through its main rhetorical components (discourse segments), hierarchically organized as its discourse tree. A similar conjecture may be stated in terms of the corresponding *finite clause-level tree* (or certain subtrees), as a hierarchically organized tree (or graph) of the text enclosed events.

As one can see from Figures 4.1.1.–4. and examples Ex. 4.1.1.–2., the clause-level trees are not necessarily embedded into the corresponding discourse segment trees: in Ex. 4.1.1., a subclause phrase makes a unitary segment with clauses in the following sentence(s), and in Ex. 4.1.2., a subclausal phrase (a non-finite clause) is broken (detached) and adjoined to the next clause, making a discourse segment.

Making comprehensible the (global) projection function is important also from another (somehow surprising) point of view: *anaphora resolution*. *Referring* an individual (object or person), a process (event or existence), or a whole bunch of actions that corresponds to a larger text span is equivalent to equating the referee category as the value of the '*inverse*' *of linguistic projection function* applied to the corresponding text. In other words, a phrase that points at a specific text span would be naturally associated with the *head* (or *nucleus*, kernel, projection tree, or another linguistic object) that is (locally or globally) projected into that text, thus with the value of the *projection function inverse* applied to that text. This perspective shows complementary facets of the linguistic projection mechanism and its specification.

Two simple examples may give a better idea of the approach we propose: "*Plecarea vânătorilor în munţii Călimani pe o vreme atât de rea a fost pe nepregătite.* Aceasta *le-a fost fatal.*" The demonstrative pronoun "*Aceasta*" refers to the whole previous sentence, and one could associate it with the sentence (and finite clause) predicate head "*a fost pe nepregătite*", or even corroborate it with the predicational head "*plecarea*" of the enclosed non-finite clause. These phrases represent "*inverses*" of the projection function, applied to the whole sentence at the local, clause-level.

Another possible example is to associate the phrase that refers a whole story within a (larger) text span to the discourse or clause-dependency tree of that text, *i.e.* to the value of '*inverse*' *projection function*, applied to that text, at the global level. This correspondence relates the story reference expression to specially computed nodes and/or subtrees in the mentioned trees.

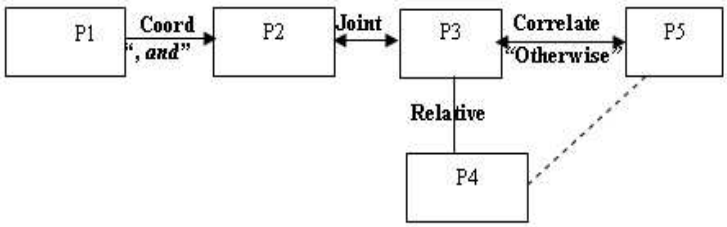


Figure 4.1.1. Inter-clause tree inherent to the segment tree of Ex. 4.1.1.

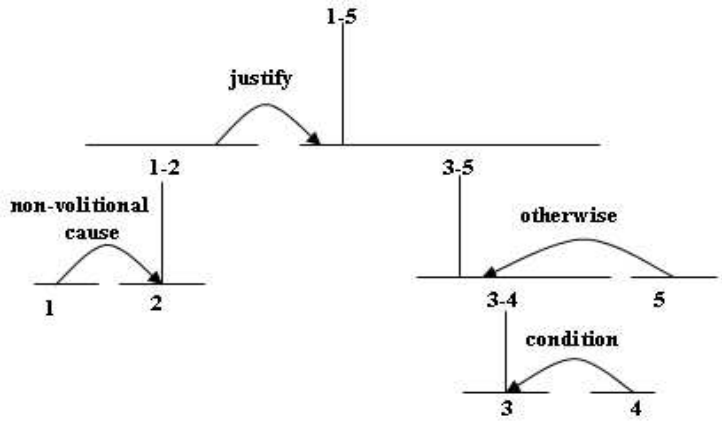


Figure 4.1.2. The RST segment tree for Ex. 4.1.1. [24; Fig. I-13]

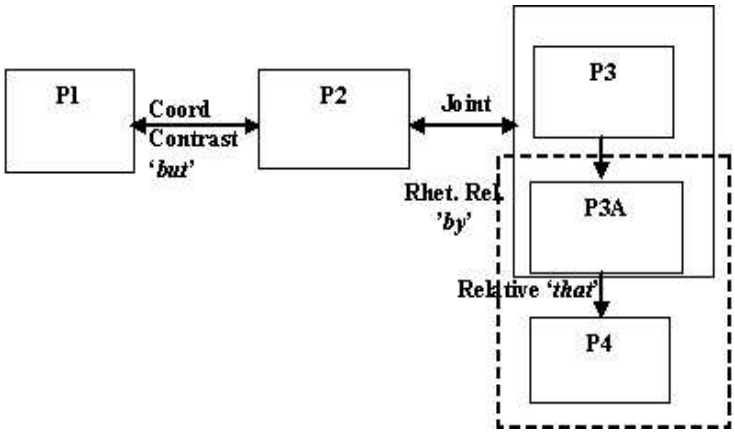


Figure 4.1.3. Clause-level tree inherent to the segment tree of Ex. 4.1.2.

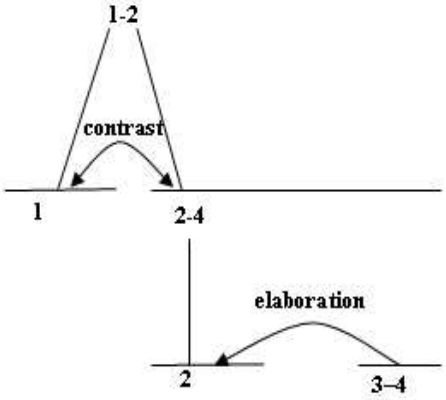


Figure 4.1.4. The RST segment tree for Ex. 4.1.2. [24; Fig. I-19]

The problems of *linguistic projection* at the *global level*, floors 3 and 4 in DFX-bar scheme (Fig. 2.2), are especially complex. Two (counter)examples show that an RST discourse segment is not necessarily the projection of its enclosed *saturated matrix clause,* as one would expect. Corroborated with the fact of subclausal discourse segments [14], [17], this gives the flavour for the difficulty of the problems for specifying the *discourse* (*global*) *projection function*, as well as *its* '*inverse*' one, *i.e. the nucleus* (or *head*) *structure* whose projected value is a certain (larger or smaller) text span.

Ex. 4.1.1. [24; p.68] **(1)** *Un nou număr din broşură este în curs de apariţie,* **(2)** *şi acest lucru înseamnă o şansă pentru noi propuneri de proiecte.* **(3A)** *Oricine* **(4)** *doreşte să actualizeze intrările în broşură* **(3B)** *ar trebui să aibă copia până la 1 Decembrie.* **(5)** *În caz contrar va fi utilizată intrarea existentă.*

[24] notices that, for the rhetorical relations **condition** and **otherwise**, their classical constructions for RST diagrams do not cover the text (similar to the classical programming) conditional "*If* A, *then* B. *Otherwise* C". These syntactic constructions receive a special attention in the latest version of SCD, falling under the (important) category of *correlated constructions* (and *clauses in correlation*) [17] (see Subsection 8.1, Part II).

Ex. 4.1.2. [24; p.76] **(1)** *Animalele se vindecă,* **(2)** *dar arborii se compartimentează.* **(3)** *Ei rezistă întreaga viaţă la răni şi infecţii* **(4)** *prin instalarea unor graniţe care rezistă la extinderea microorganismelor invadatoare.*

The figures 4.1.1-4.1.4 showing the inter-clausal relations and discourse trees for the texts in Ex.4.1.1–2. support our statements concerning the global projections at these clause and discourse levels.

4.2 A Special Case: Sub-Clausal Discourse Segments

The essential difference between Marcu's discourse segmentation [25], [26], and the SCD syntax-driven segmentation is the type of target structures that the two algorithms are looking for: Marcu's algorithm's

objective is to obtain structures derived from RST rhetorical relations [25], while SCD's main purpose is to reveal the sentential, syntactic-semantic structures, at the sentence level, from syntactic category-headed phrases and non-finite clauses, to finite clauses and inter-clausal (syntactic and logical-semantic) relations. Between rhetorical relations and inter-clausal relations of syntactic-semantic nature there is a subtle, distinctive, however close relationship. The following examples from [25; Appendix A] illustrate some aspects of this situation:

Ex. 4.2.1. [25; Text A.4, p. 268] [*Every rule has exceptions,*] [*but the tragic and too common tableaux of hundreds or even thousands of people snake-lining up for any task with a paycheck illustrates a lack of jobs,*] [*not laziness.*]

Comments. The discourse segment [*not laziness.*] is actually a clause, defective of its (finite) predicate "*illustrates*".

Ex. 4.2.2. [25; Text A.6, p. 269] [*Cleaning agents on the burnished surface of the Ectype coating actually remove build-up from the head,*] [*while lubricating it at the same time.*]

Observation. The situation when an elementary discourse unit (EDU, or discourse segment) is properly embedded into a (finite) clause is very close to that when a discourse marker splits a finite clause into two spans, each span belonging to distinct EDU*s*. This does not necessarily mean that the two EDU*s* are both enclosed into the same finite clause; the most frequent situation is when a discourse segment tears a phrase from a clause and continues its span on the next clauses(s).

These examples bring further arguments that the relationship between the *discourse segment tree* and its underlying *clause-level tree* is an intricate interplay, falling both in the field of lexical semantics for *local structures* but especially in the area of clause-level and discursive semantics for *global structures* of the text organization. Segment tree projection function is closely related to the composition between the discourse marker semantics and the clause-level predications involved by the subsumed clauses. Until one would know more about the projection functions at global level, about the relationship between clause-dependency and discourse trees of a text span, and how the global projection functions and their inverses work, these issues remain

still open and challenging problems.

5 Transitory Conclusions

The *phrase markers* play a fundamental role in delimiting the syntactic (and also, semantic) structures and establishing their dependencies. I emphasized this since the beginnings of SCD ([7] and earlier). One can see now a whole movement toward rediscovering the essential role of markers, especially on the *discourse* and higher levels of the text. The SCD linguistic strategy, in particular the local and global (D)FX-bar theory, is trying to use and to put to work not only the '*connectives*' of several types, '*cue phrases*', '*discourse markers*', etc. but the *whole palette* of markers, for *all* the (local and global) levels of analysis-generation of NL, from lexical to discourse ones, especially *in the syntax*. SCD makes a special effort to maximize the use of the lexical-semantics and syntactic means in discovering the logical-semantics and discursive structures of NL.

The main novel aspects that make the difference between (D)FX-bar schemes and previous X-bar type theories may be summarized as follows: **(a)** the two *global* FX-bar schemes that represent extensions to the *clause* and *discourse levels*, both enclosing an improved shape of the same *local* FX-bar *scheme*; **(b)** the *graph-based hierarchy* of the SCD *marker classes* that are used in the FX-bar general schemes, for the local and global levels; **(c)** theoretical arguments supporting a *lexical-level predication*, based on the *predicational feature* assignment to the major lexical categories N, V, A, since the lexicon description; **(d)** local and global FX-bar projection problems, with emphasis on the fundamental VG (and VGK) local structures; **(e)** The maximal use of the functional (predicational) and relational features of the local and global markers represent an adequate framework for defining the concept of *functional generative capacity* (in Part II), with interesting consequences on the design and taxonomy of local / global text segmentation / parsing algorithms.

DFX-bar scheme may be associated also to a *language-dependent automaton* (working similarly, however, for a large class of NL*s*) that

starts with a sentence, receives *on-line* each word of it, and stops at the final punctuation sign. For adequate values of the parameters like *word* (*argument*) *ordering* and *projection direction* of the major categories and markers, the FX-bar scheme can properly represent the correct dependency of linguistic structures.

As a basic component of the SCD linguistic strategy [9], the *local and global* FX-bar *theory* may also be seen as a *procedural mechanism* providing a consistent set of principles and rules that ensure a sound functioning of the FX-bar *schemes*, from the lexicon to the discourse level organization of the NL analysis / generation processes. Continuing this perspective and paraphrasing A. Joshi's well-known *tree adjoining grammar* (TAG) [21], SCD strategy may further be understood as a *theory* of (D)FX-bar *schemes* (thus *tree*) *checking and adjoining* (see subsection 3.1). The same role of procedural mechanism for FX-bar scheme(s) is envisaged for the related but more general model of *Marcus contextual grammars* [33], as a down-to-language strategy putting to work (highly)-*contextual mechanisms* (such as SCD marker classes and dependency principles) for the NL phrase structure recognition and generation. These are just samples of the role that FX-bar theory can still really play within the NL theory and technology.

The global (D)FX-bar scheme exposed in Fig. 2.2 represents an essential extension to the global approach in the context of SCD linguistic strategy. Each of major lexical categories X = N, V, A, along with the grammatical category CL and the discourse category SEG, are projected (recursively) on *three* bar levels (BAR = 0, 1, 2), within *five* local-global levels of FX-bar linguistic projection process. All these structures, except the lexicon normalized X0-lex form, are functionally and/or relationally "marked", through multiple applications, by the *four*-level local / global markers (on the first level of the hierarchy, followed by other sub-hierarchies) whose classes are better specified within the SCD linguistic strategy (Section 7, Part II).

Functional properties of the (predicational or relational) categories can be assigned even from the lexicon level, but the semantic and/or pragmatic context may entail temporarily loosing or gaining such a quality. This is also true for phrases and collocations resulting from

the lexical analysis. Discovering and pointing out the *functional* (*functorial*) and *relational properties* of the words and phrases is an essential task of the NL *parsing* (analysis and generation) processes; this is specific not only to SCD linguistic strategy, but also to *principle-based parsing* strategies, *e.g. rhetorical parsing* [25]. The proposed FX-bar schemes, consolidating the basic ideas of AX-bar schemes [7], [9] and FX-bar theory [11], [12], [16], provide both a theoretical and practical tool for local and global parsing / generation text processing tasks.

A central issue for obtaining a solution to the *direct* and *inverse* FX-bar *projection problems* of VG (VGK) consists in defining and computing *diathesis transformations* and *semantic diathesis* functions, showing that these function values may characterize the way VG structure is (reversely) FX-bar projected into its (predicational) semantic head, as well as (directly) FX-bar projected, on the lines of its semantic head meaning(s), into the corresponding clause(s). This mechanism is described in [18], supporting a better understanding of *lexical predication* anatomy and functioning.

References

[1] Barbu, Ana-Maria; Emil Ionescu (1996): *Contemporary grammatical theories*: *grammars of the phrasal head.* in Limba româna, no. 45, (1-6) pp. 31–55 (in Romanian).

[2] Barbu, Ana-Maria (1999): *The Verbal Complex.* Studii si Cercetari Lingvistice, L, no.1, Bucureşti, p. 39–84 (In Romanian).

[3] Chomsky, Noam (1981): *Lectures on Government and Binding.* Foris, Dordrecht.

[4] Chomsky, Noam (1986): *Barriers.* The MIT Press, Cambridge.

[5] Chomsky, Noam (1995): *The Minimalist Program.* The MIT Press, Cambridge, Massachusetts.

[6] Cristea, D., O. Postolache, I. Pistol (2005): *Summarisation through Discourse Structure.* In Proceedings of CiCling 2005, Springer LNSC, vol. 3406.

[7] Curteanu, Neculai (1988): *Augmented X-bar Schemes.* COLING'88 Proceedings, Budapest, pp. 130–132.

[8] Curteanu, Neculai (1990): *A Marker-Hierarchy-based Approach Supporting the SCD Parsing Strategy.* Research Report no. 18, Institute of Technical Cybernetics, Bratislava, Slovak Republik.

[9] Curteanu, Neculai (1994): *From Morphology to Discourse Through Marker Structures in the SCD Parsing Strategy. A Marker-Hierarchy Based Approach.* Language and Cybernetics, Akademia Libroservo, Prague, pp. 61–73.

[10] Curteanu, Neculai; G. Holban (1996): *SCD Linguistic Strategy Applied to the Analysis and Generation of Romanian.* In (Dan Tufiş, Ed.) Language and Technology, Romanian Academy, Bucharest, pp. 169–176 (in Romanian).

[11] Curteanu, Neculai (2002): *Elements of a Functional X-bar Theory Within the SCD Linguistic Strategy*, ECIT2002 Conference, Iaşi, România.

[12] Curteanu, Neculai (2003): *Towards a Functional X-bar Theory.* In the volume "The Romanian Language in the Informational Society", Dan Tufiş, F. Filip (Eds.), Edited by the Romanian Academy, Research Institute for Artificial Intelligence, Bucharest, pp. 51–86 (in Romanian).

[13] Curteanu, Neculai; D. Gâlea; C. Linteş (2003): *Segmentation Algorithms for Clause-Type Textual Units.* In the volume "The Romanian Language in the Informational Society", Dan Tufiş, F. Filip (Eds.), Edited by the Romanian Academy, Research Institute for Artificial Intelligence, Bucharest, pp. 165–190 (in Romanian).

[14] Curteanu, Neculai; D. Gâlea; C. Butnariu; C. Bolea (2004): *Marcu's Clause-like Discourse Segmentation Algorithm and SCD Clause Segmentation-based Parsing*, In the volume "Intelligent Systems" (Ed. Horia-Nicolai Teodorescu). Selected Papers from ECIT2004 Conference, Iaşi, România, pp. 59–86.

[15] Curteanu, Neculai (2003-2004): *Contrastive Meanings of the Terms* "Predicative" *and* "Predicational" *in Various Linguistic Theories* (I, II). Computer Science Journal of Moldova (R. Moldova), Vol. 11, No. 3(33), 2003 (I); Vol. 12, No. 1(34), 2004 (II).

[16] Curteanu, Neculai (2005): *Functional FX-bar Theory Extended to Discourse (Rhetorical) Structures.* In 'Intelligent Systems' Conference Volume, H.-N. Teodorescu *et al.* (Editors), Performantica Press, Iaşi (Romania), pp. 169–182.

[17] Curteanu, Neculai; E. Zlavog; C. Bolea (2005): *Sentence-Based and Discourse Segmentation / Parsing with SCD Linguistic Strategy.* In 'Intelligent Systems' Conference Volume, H.-N. Teodorescu *et al.* (Editors), Performantica Press, Iaşi (Romania), pp. 153–168.

[18] Curteanu, Neculai; Diana Trandabăţ (2006): *Functional (F)X-bar Projections for Local and Global Text Structures. The Anatomy of Predication.* Revue Roumaine de Linguistique, Bucharest (to appear).

[19] Dobrovie-Sorin, Carmen (1994): *The syntax of Romanian. Comparative Studies.* Berlin: Mouton de Gruyter.

[20] Irimia, Dumitru (1997): *The Morphosyntax of the Romanian Verb.* The Editorial House of the "Al. I. Cuza" Iaşi University (in Romanian).

[21] Joshi, Aravind K. and Ives Schabes (1997): *Tree Adjoining Grammars.* In "Handbook of Formal Languages and Automata" (A. Salomaa *et al.*, Eds.), Vol. 3, Heidelberg, Springer-Verlag.

[22] Kornai, András, Geoffrey Pullum (1990): *The X-bar Theory of Phrase Structure*, Language, Vol. 66, No. 1, pp. 24–50.

[23] Legendre, Géraldine (1999): *Optimal Romanian clitics: a cross-linguistic perspective.* In: V. Motapanyane (Ed.) Comparative Studies in Romanian Syntax. HAG, The Hague.

[24] Mann, William, Sandra Thompson (1988): *Rhetorical Structure Theory: A Theory of Text Organization.* Research Report RS-87-190, Information Sciences Institute, University of Southern California, Marina del Rey, California, 80 pp.

[25] Marcu, Daniel (1997): *The Rhetorical Parsing, Summarization, and Generation of Natural Language Texts*, Ph.D. Thesis, Univ. of Toronto, Canada, pp. 341.

[26] Marcu, Daniel (2000): *The Theory and Practice of Discourse Parsing and Summarization.* The MIT Press.

[27] Miller, Philip (1999): *Strong Generative Capacity. The Semantics of Linguistic Formalism.* CSLI Publications, Stanford, California, 1999.

[28] Monachesi, Paola (1998): *The Morphosyntax of Romanian Cliticization.* In: P.-A. Coppen *et al.* (Eds.), Proceedings of Computational Linguistics in The Netherlands 1997, pp. 99–118, Amsterdam-Atlanta: Rodopi.

[29] Monachesi, Paola (2000): *Clitic placement in the Romanian verbal complex.* In: B. Gerlach and J. Grijzenhout (eds.) Clitics in phonology, morphology, and syntax. Linguistik Aktuell. John Benjamins. Amsterdam.

[30] Monachesi, Paola (2005): *The Verbal Complex in Romance. A Case Study in Grammatical Interfaces.* Oxford University Press, Oxford Studies in Theoretical Linguistics.

[31] Orasan C. (2000): A hybrid method for clause splitting in unrestricted English texts Available at: http://www.wlv.ac.uk/sles/compling/papers/orasan-00.pdf

[32] Passonneau, Rebecca; Diane Litman (1997): *Intention-based segmentation: human reliability and correlation with linguistic cues*, in Proc. 31th Annual Meeting of ACL, Ohio, pp. 148–155.

[33] Gheorghe Păun: *Marcus Contextual Grammars*. Kluwer Academic Publishers, Dordrecht, 1997.

[34] Pollard, Carl; Ivan Sag (1994): *Head-Driven Phrase Structure Grammar*. The University of Chicago Press, Chicago & London.

[35] Popârda, O.; N. Curteanu (2002): *L'évolution du discours juridique français analysé par la stratégie linguistique SCD*. In the volume "Representations du Sens Linguistique", Lagorgette, P. Larrivée (Eds.), LINCOM Europa, series Studies in Theoretical Linguistics, München, Germany, pp. 487–502.

[36] Puşcaşu, G. (2003): *Elementary discourse unit segmentation*. Dissertation thesis. "Al.I.Cuza" University of Iasi.

[37] Sgall, Petr; E. Hajičova, J. Panevova (1986): *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. Kluwer Academic Publishers, Dordrecht.

[38] Soricut, R. and Daniel Marcu (2003): Sentence Level Discourse Parsing using Syntactic and Lexical Information. *Proceedings of the Human Language Technology and North American Association for Computational Linguistics Conference (HLT/NAACL)*, May 27-June 1, Edmonton, Canada.

N. Curteanu, Received February 21, 2006

Institute for Computer Science,
Romanian Academy, Iaşi Branch
B-dul Carol I, nr. 22A, 6600 IAŞI, ROMÂNIA
E–mail: *ncurteanu@yahoo.com* and *curteanu@iit.tuiasi.ro*

Computer Science Journal of Moldova, vol.**14**, no.1(40), 2006

The ascertainment of the inflexion models for Romanian*

Svetlana Cojocaru

Abstract

A method to increase the degree of inflexion process automatization for Romanian is proposed.

1 Introduction

The problem of the automation of words inflexion process in Romanian was investigated in [1], [2]. The obtained results permitted to construct a computational lexicon containing about 1 million of words: the lemmas and their word-forms. The inflexion process was based on two methods: a static and a dynamic ones. The first one is operating with a morphological dictionary[3], where the inflexion group is indicated explicitly; the second method tries to find the inflexion model analysing the word's structure, especially the affixes series. These series were determined by examination of vocabularies from different lexicographic sources. The dynamic method was implemented as an interactive program, which is able to inflect automatically about 80% [4] of words. The usage of this program shows that a user intervention is requested often to solve some ambiguities, although those cases could be solved automatically. In this paper we intend to improve the dynamic method in order to increase its degree of automation. In the first section we recall the definition of inflexional grammars with scattered context [1], in the next two sections the inflexion criteria are analyzed and an algorithm to determine the inflexion model for a given word is proposed.

*Supported by INTAS project Ref. Nr. 05-104-7633

2 Scattered context grammars for vocabulary generation

The starting point for this approach was the book [3], where main part of Romanian inflective words were classified according to the methods of the word-forms creating. There were 100 groups for masculine nouns, 273 - for verbs, etc in the book, and about 30000 words with their group numbers were listed. The classification was made from the linguistic point of view, and, for example, the accents were taken into account. In our case we can operate only with the graphical representation of the word, what equally simplifies and complicates the problem. Nevertheless, this classification was useful and have lead to the idea to introduce the special grammar to formalize word-forms producing.

Definition 1 *The object $G = \{R, T, *\}$, where R is the set of rules, T is the (ordered) set of the list of endings, $*$ is a special symbol not contained in any words of the given language, is named an inflexion grammar.*

The grammar rules have the following form:

$$[/]^*[\#][N_1]a_1\overline{b_1}a_2\ldots a_{n-1}\overline{b_{n-1}}a_n \longrightarrow a_1^{'}\overline{b_1}a_2^{'}\ldots a_{n-1}^{'}\overline{b_{n-1}}a_n^{'}\ N_2,$$

where a_i , $a_i^{'}$ are arbitrary words and either b_i is nonempty word or the special symbol $*$ stands instead of $\overline{b_i}$. N_j — endings set numbers.

The interpretation of this rule is as follows. Let w be the word to produce word-forms (basic word-form). Every sign / indicates cutting the last letter from w. The obtained (after the deletions) word v is considered as a root (if N_1 exists) and N_1 is its index in endings sets list L. In any case the word v should have the form

$$f_0a_1f_1a_2f_2\ldots a_{n-1}f_{n-1}a_nf_n,$$

where every f_i is arbitrary (possible empty) word, not containing (for $i = 1, 2, \ldots, n-1$) the veto subword b_i. If there exists more then one representation of this kind the first (scanning v from left to the right

or vice versa if the sign # is present) should be selected. The special character $*$ instead of $\overline{b_i}$ admits arbitrary f_i.

After the evident substitution the word $f_0 a_1^{'} f_1 a_2^{'} \dots a_n^{'} f_n$ serves as a second (or first, if N_1 is absent) root and N_2 is its endings set number.

Veto for b_i is conditioned by the necessity to determine the position of the subword a_i to be substituted.

Using these grammar rules, we can formalize the process of creating of the decomposed vocabulary. According to the classification in [3], it is possible to build the grammar rules for every group. Sometimes more than two roots arise and more than one grammar rule is necessary.

The inflexion grammar for Romanian contains 866 rules and 320 endings sets. They were used to obtain a morphological dictionary with about 30000 basic lemmas.

3 Automatic inflexion criteria

The grammar rules define, in fact, the inflexion model on the algorithmic level: cutting a given number of symbols at the word ending, obtaining different roots by means of (parallel) substitutions (in order to produce vowel and consonant alternation), attaching the corresponding endings to the roots. But this method can be applied only to the case, when we know the inflexion group number. If this number is unknown the problem to ascertain the inflexion model having the graphical representation of the word arises. Is it possible to solve this problem algorithmically? The answer is a negative one. The first impediment is to determine the part of speech: there are a lot of homonymies denoting different parts of speech (Example: *abate* – a masculine noun and a verb. In the first case it means "abbot" and "to divert" in the second).

We can restrict the formulation of the problem: is it possible to determine the inflexion model (respecting the conditions mentioned above) if we know the part of speech? The answer is a negative one in this case also. For confirmation one can adduce a list of examples which prove that the ascertainment of the inflexion model is impossible if we don't invoke the phonetic or etymological information. Let us see

only one example of this kind: the feminine noun *masă*. Following the meaning "table" the plural will be formed as *mese*, using the model with vowel alternation "$a \rightarrow e$". But if we follow the meaning "mass" the plural *mase* will be obtained without any alternation. The origin of this phenomenon is an etymological one: in the first case the word derives from the Latin *mensa*, in the second case the French *masse* precedes it [5].

But the problem might be tackled in another way: to establish some criteria which permit after the analysing of the word structure to conclude about the possibility to determine the inflexion model and, if this is possible, to fix the specific model. Otherwise, we will try to formulate the criterion according to which one can affirm that the inflexion process can be performed automatically and denote the corresponding model.

Thus, let we have a word (a lemma) in its graphical representation. We know the part of speech, and the gender in the case of nouns. We will divide all words into three categories: irregular, absolute regular and partial regular.

For each part of speech the belonging to the group of the irregular words is determined by its belonging to the set of words, picked apriori. We will consider absolutely regular the words which admit the automatic inflexion. We will call partially regular the words which need some additional information (except the graphical representation) to be inflected. In the next section we will formulate the criteria of the belonging to the last two groups and establish the corresponding inflexion models.

3.1 The algorithm of the inflexion model ascertainment

Let $CG = \{M, F, N, A, V, P\}$ be the set of grammar categories which denote masculine, feminine and neuter nouns, adjective, verb and pronoun respectively. Let $c \in CG$ and GF be an inflectional grammar. We will denote by L_c the list of pairs (α, μ), where α is a word of category c, and μ is its corresponding inflexion group number. Two inflexion groups μ_1 and μ_2 will be considered equivalent if they have the

same corresponding set of grammar rules from the inflectional grammar GF. To simplify the explanation the set of irregular words will be excluded from the examination; their presence or absence doesn't affect the generality of the algorithm.

Let us denote as $N_{max} = max \mid \alpha \mid$ the maximal length of the words $\alpha \in L_c$. Let $A_j = \{a_{1j}, a_{2j}, \ldots, a_{kj}\}$ be the set of endings with length j of words α ($j \leq N_{max}$). We will denote by n the length of the current ending. For each inflexion group μ we will put in correspondence a set S_μ of endings, which is initially empty. The equivalent groups will have the same corresponding set.

1. $n := 1$
2. $i := 1$
3. Select all the words containing the ending $a_{in} \in A_n$. For each of them we fix its inflexion group μ.
4. If all the inflexion groups are equal or equivalent we include the ending a_{in} into the set S_μ, exclude from the list L_c the words with ending a_{in} and go to step 6.
5. If the selected words have different (nonequivalent) groups do the following verifications:
   - the ending $a_{in} = \alpha'$ and there are the pairs (α', μ_1) and $(\alpha', \mu_2) \in L_c$. In this case the word α' is included into the partially regular category;
   - the ending $a_{in} = \alpha'$ and there are the pairs (α', μ_1) and (α'', μ_2), where $\alpha'' = \beta\alpha'$. In this case the word α' is included into the partially regular category.
6. Increment i by 1 ($i \leq k$) and repeat the process from the step 3. If $i > k$ increment n by 1 and follow step 2. The process will finish when $n > N_{max}$.
7. Construct the union of the sets having the same corresponding grammar rule.

The obtained result constitutes the set of automatic inflexion criteria.

3.2 Example of the algorithm application

We will illustrate the algorithm functioning applying it to the list of masculine nouns from [3]. The list contains about 5000 words. A fragment of it (where we added the corresponding English translations) looks as following:

abur	M1	(steam)
leuşor	M2	(little lion)
abonat	M3	(subscriber)
watt	M4	(watt)
brad	M5	(fir)
urs	M6	(bear)
boss	M7	(boss)

.....

We will operate with the inflexion grammar GM. A part of it is presented below:

M1 1;
M2 2 $u \longrightarrow$i 3;
M3 2 $t \longrightarrow$ ţ 3;
M4 2 $tt \longrightarrow$ ţ 3;
M5 2 $d \longrightarrow$ z 3;
M6 2 $s \longrightarrow$ ş 3;
M7 2 $ss \longrightarrow$ ş 3;
. . . . . .

The grammar rules are refering to the following paradigms:

```
1 [- - - ul ului ule i i i ii ilor ilor]

2 [- - - ul ului ule]

3 [i i i ii ilor ilor]

. . . . . .
```

The algorithm application generated the sets of endings, which ascertain the inflexion groups. We present here a part of them:

$a_f \in$ {b} $\cup$ {ic, ec, rac, mac, bac, âc, uc, dac, oc, nc, lac, zac, vac, rc, lc, geac, tac, lac, nac, pac, sac, jac, şac, cac} $\cup$ {fag, arag, ărag, bag, mag, ng, og, ug, ig, eg, rg, lg} $\cup$ {f} $\cup$ { h} $\cup$ {j} $\cup$ {ul, ol, ăl, ll, şial, cial, til, cil, mil, fil, ril, bil, vil, dil, xil, zil, nil, hil, upil, ral, tal, fal, şal, ibal, nal, lal, mal, pal, gal, dal, ual, val, sal, ghel, fel, udel} $\cup$ {m} $\cup$ {mn, en, in, on, ân, rn, un, vn, gan, can, zan, ban, nan, san, ran, tan, lan, van, han, pan, dan, ţan, uan, fan, aolean, oman, aman, rman, iman, esman, osman, hman, bman, şman, atman, lman, dman, rmăn, badian, radian} $\cup$ {ţap, up, ip, op, rp, mp, ep, cap, sap, rap, lap, nap} $\cup$ {ur, or, ir, âr, rr, ier, ger, mer, per, ler, her, fer, ber, xer, ner, ter, der, zer, jer, ţer, ier, ser, rer, ver, şer, găr, şafăr, tăr, hăr, căr, văr, bar, car, dar, far, ear, gar, har, iar, jar, mar, par, rar, sar, tar, oar, ţar, şar, var, zar, tuar, iuar, ouar, guar, zuar, onar, inar, unar, snar, enar, tnar, arnar, rnar, ânar, gnar, mnar, znar, olar, elar, ilar, glar, ular, blar, slar, plar, bălar, tănar, lănar, omănar, cănar, iermănar} $\cup$ {v} $\cup$ {ez, onz, lz, baz, ăz, âz, ruz, ăuz, moz, guz, tz, muz, suz, luz, iz, mz, anz, laz, uoz, tuz} $\cup$ {ş} $\cup$ {eţ, uţ, nţ, iţ, căţ}$\rightarrow M1$.

$a_f \in$ {it, ot, pt, ct, lt, ut, et, rt, şt, ft, ît, ent, ant, int, ont, unt, sânt, nat, tat, lat, bat, mat, zat, gat, pat, jat, rat, cat, sat, vat, eat, oat, ţat, fat, dat, şat, niat, liat, ciat, uiat, ţiat, miat, giat, diat, ariat, triat, priat } $\rightarrow M3$,

$a_f \in$ {tt} $\rightarrow M4$,

$a_f \in$ {d} $\rightarrow M5$,

$a_f \in$ {os, es, as, us, is, âs, ns, ps, rs, cs} $\rightarrow M6$,

$a_f \in$ {ss} $\rightarrow M7$

If the ending a_f of the word w belongs to one of the mentioned above sets, then it can be inflected according to the grammar rules which correspond to the indicated inflexion group.

The following endings point to partially regular nouns:

$p_f \in \{\text{osc}\} \to M17, M18;$
$p_f \in \{\text{iac}\} \to M13, M39;$
$p_f \in \{\text{drag}\} \to M14, M15;$
$p_f \in \{\text{gaci}\} \to M73, M98;$
$p_f \in \{\text{opil, cal, bel, ocel}\} \to M1, M12;$
$p_f \in \{\text{rial}\} \to M1, M43;$
$p_f \in \{\text{bouşor, cer}\} \to M1, M2;$
$p_f \in \{\text{leat}\} \to M3, M31;$
$p_f \in \{\text{lustru, leandru}\} \to M62, M63;$
$p_f \in \{\text{iandru}\} \to M62, M65;$
$p_f \in \{\text{roz}\} \to M1, M29.$

If an ending of a masculine noun belongs to the endings set a_f one can affirm, that its declination can be performed according to the grammar rules which correspond to this set. If the ending belongs to a set p_f than we can't indicate the unique model of inflexion and need some additional information to perform declination. For example, in the inflexion program [2] the user is asked to select one from the several possible word–forms of plural. This information is sufficient to fix the appropriate inflexion model. If the ending doesn't belong to any of the sets a_f or p_f, and the word doesn't belong to the list of irregular words it remains to find other methods to produce the corresponding word-forms.

The obtained result was verified on the set of about 2000 masculine nouns from [5], which doesn't intersect the set of masculine nouns from [3]. We have seen the complete correctness in the cases when the ending belongs to the sets a_f or p_f. At the same time we have found about 3% of nouns whose endings were not included into the sets generated by the described algorithm.

Conclusions and results

The automatization of the inflexion process is one of the problems which appear on computational lexicons constructing. It is especially difficult for high inflectional languages to which the Romanian one belongs as well. We have elaborated two methods to solve this problem:

a static and a dynamic one. The second one can be substantially improved applying the algorithm stated below. A computational lexicon for Romanian containing about 1 mln. words (obtained by inflexion of 60 000 lemmas) was constructed using these methods. The lexicon was used for different linguistic applications: the spelling checker for Romanian [6], the data base of linguistic resources [7], the search algorithm for web pages [8].

References

[1] S.Cojocaru, M.Evstunin, V.Ufnarovski. Detecting and correcting spelling errors for Romanian language. Computer Science Journal of Moldova, Vol.1, N.1,1993, Kishinev, pp.3–22.

[2] E.Boian, A.Danilchenco, L.Topal. The automation of speech parts inflexion process. Computer Science Journal of Moldova. 1993, Vol. 1, N.2, pp.14–26

[3] A.Lombard, C.Gâdei. Dictionnaire morphologique de la langue roumaine. Bucuresti, Editura Academiei, 1981, 232 p.

[4] The inflexion regularities for the Romanian language.Computer Science Journal of Moldova, Vol.4, N.1, 1996, Kishinev, pp.40–58

[5] Dictionarul explicativ al limbii romane. Academia Romana, Institutul de Lingvistica "Iorgu Iordan", Editura Univers Enciclopedic, 1998, 1192 p.

[6] S. Cojocaru: Romanian Lexicon: Tools, Implementation, Usage. In: Dan Tufis and Poul Andersen (eds.), Recent Advances in Romanian Language Technology. Editura Academiei, 1997, pp. 107–114.

[7] E. Boian, C. Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahova. Lexical resources for Romanian. Memoriile ştiinţifice ale Academiei Române, Bucharest, Romania, 2004, pp.267–278

[8] O.Burlaca, S.Cojocaru, C.Gaindric. A content management system for electronic theses.Proceedings of the 4rd International Conference on Microelectronics and Computer Science. Vol.II, 2005, pp.509–513.

S.Cojocaru, Ph.D.,

Received March 30, 2006

Institute of Mathematics and Computer Science,
Academy of Sciences, Moldova
5, Academiei str., Chişinău,
Moldova, MD 2028
e-mail: *sveta@math.md*

Computer Science Journal of Moldova, vol.**14**, no.1(40), 2006

Intonational Structures in Romanian Yes-No Questions

Vasile Apopei, Doina Jitcă, Adrian Turculeţ

Abstract

This paper presents the conclusions resulted from an intonational analysis of Romanian Yes-No questions. The recent analysis results consist in dividing and structuring the F0 curves into intonational units. Each intonational unit is described by a tone sequence using ToBI labels used in annotation of the most important phonetic events: pitch accents and boundary tones. The authors of the present study propose a description of the resulted patterns for F0 contour in terms of intonational units structures described by their tone sequence. We consider this description suitable for the variety of melodic contours resulted from different speakers and different focalizations in their utterances. In paragraph 3, the paper presents the intonational variants resulted from our speech corpus analysis. The conclusions of Yes-No question analysis are important for linguistic studies and in Romanian speech synthesis.

1 Introduction

This paper presents the conclusions resulted from an intonational analysis of Romanian Yes-No questions. This study continues our previous works, the results of which were published in [4], [5], [6]. In the previous study [6] we performed a phonetic and auditory analysis for the "neutral" utterances, only (uttered without an intention to focus certain words) and we presented a description of their intonation based on the ToBI annotation system.

In the present work we are interested both by the utterances "without focus" and those "with focus" on different words of the sentence. The recent analysis results consist in dividing and structuring the F0 curve into intonational units. Each intonational unit is described by a tone sequence using ToBI labels used in annotation of the most important phonetic events: pitch accents and boundary tones [3]. These results are comprised in a series of tables, which contain in each row the description of an intonational variant. The phonological and semantic analysis of Yes-No questions implies the establishment of interrogative emphasis position and, consequently, this information is contained in these tables.

Laurenţia Dascălu-Jinga, in her study on the patterns of Romanian melodic contours, presents conclusions about the patterns of final contours and the positions of interrogative emphasis in the case of Yes-No "neutral" questions [1].

The authors of the present study propose a description of the resulted patterns for F0 contour in terms of intonational units structures, each of them being described by their tone sequence marked by the labels of ToBI annotation system. This description modality of the F0 patterns is valid both for "neutral" and "non-neutral" cases of the Yes-No questions.

We consider this description suitable for the variety of melodic contours resulted from different speakers and different focalisations in their utterances.

The F0 contour was divided into intermediate phrases. In this paper, we denote both intermediate and intonational phrases by the term "intonational unit". In paragraph 3, the paper presents the intonational variants resulted from our speech corpus analysis. The speech corpus is generated by a methodology presented in the next paragraph.

2 The methodology for building the speech corpus

In generating the speech corpus we chose three types of sentences to be uttered, with the final word being oxytone, paroxytone and, respec-

tively, proparoxytone. In the corpus we had the intention to generate utterances with various final F0 contour patterns. The texts are the following:

Ai văzut acest ***afiş****? (Did you see this* ***poster****?)*

Ai văzut afişul ***acesta****? (Did you see* ***this*** *poster?)*

Ai văzut regele? (Did you see the king?)

These texts were uttered by speakers who live in different regions of Romania: from Moldova (speakers AT and LS), south-west Transilvania (speakers LM and IM) and Banat (speaker TM). Each speaker generated 5 "neutral" utterances, without any intention to focus a word, of each text corresponding to oxytone, paroxytone and proparoxytone cases.

Then, speakers uttered each text in different interpretations (five times per each interpretation): with focus on the verb, with focus on the first word of nominal group, and with focus on the second word of the nominal group. We included the focus indications for speakers, in the last column of the tables that contain pattern descriptions.

The results of F0 contour interpretation are presented in the next paragraph.

3 The phonetic and phonologic analysis of the utterances

The purpose of our phonetic and phonologic analysis is to divide the utterances into intonational units (intonational phrases or intermediate phrase) and then to annotate the tonal events on F0 curve, as generating the tone sequence for each of them. In annotation we used the ToBI label system [3].

An intonational unit represents a segment of the F0 curve that contains one primary pitch accent and, possibly other less prominent accents. The primary accents are visible either on the F0 curve, by significant F0 frequency variations, or by the energy level during one stressed syllable within the corresponding unit. The significant F0 frequency variations during the unstressed syllables following a primary

accent can represent an indication for the boundary between two consecutive units. These units can comprise one or many words linked by a meaning or by the syntactic structure.

The identification of the primary tonal accents on F0 curves entails their division into intonational units. The F0 patterns of intermediate phrases resulted from our analysis was grouped into two major categories:

- Units characterised by a F0 contour that begins with a down-stepped trend of the F0 frequency, until a low tone appears, and then finishes with an increasing segment of F0 frequency. We denote this type of unit with "A";
- Units characterised by a F0 contour that begins with a up-stepped trend of the F0 frequency, until a high tone appears, and then finishes with an decreasing segment of F0 frequency. We denote this type of unit with "B".

The definition of "A" and "B" units is a very general one. An "A" unit can be generated by different tone sequences, as the following: L* L-H%, H+!H* H* H-%, L* H-H%. The same for a "B" unit, it may be performed by different tone sequences, as the following: H* L* L-L%, H*+L L-L%.

These two basic prototypes of intonational unit patterns have different variants. In some cases an extra short segment corresponding to the final unaccented syllables (after the last pitch accent) is added. Thus, at the end of an "A" unit a fall of the tone may exist while, at the end of a "B" unit, an increase of the tone may occur. Both extra segments are characterised by a small range of the tone variation and by a short time length.

In the beginning of the sentences, within an "A" unit it can be missed the first down-stepped segment, while, in the end of the sentence, in a "B" unit the first up-stepped one may be missed.

In the case of pitch contours characterised by a small range between *Low* and *High* levels, the "A" and "B" patterns of F0 curve can not be identified because the contour is almost a flat one. In ToBI annotation

system, these segments are characterised by the value *Low* of the parameter HIF0. This kind of contours can spread over an entirely unit (we consider it as a particular case of an "A" unit) or make a segment into an intonational unit of type "A".

We claim that one or more intonational units of one type described before, linked in a sequential structure, can define the intonation of an utterance. In our perspective, the F0 contour shape of one utterance may be divided into a pattern sequence of "A" or "B" type.

The description of intonational variants are based on the following labels:

- H*, L* – for high and respectively, low monotonic pitch accents,
- L+H*, L*+H, H+L* – for bitonal pitch accents with a high or low tone in a middle of accented vowel. The tone is marked by "*";
- H+!H*- for a bitonal accent that marks a decreasing pitch during accented syllable, but not until a low tone;
- H-, L- for the phrase accent tones that mark the end of an intermediate phrase;
- %H, %L for the phrase accent tones that mark the beginning of an intonational phrase;
- H%, L% – for the boundary tones in the end of an intonational phrase;
- HIF0 – for the ToBI parameter that indicates the Topline level used in pitch range modelling. It takes the following values: L-low, H-high;
- H(*) and L(*) – for the accents H* and respectively L* marked in the case of oxytone final word units .

The diacritics "^" and "!" that precede a high tone indicate a higher level, respectively, a lower level than the previous high tone.

To introduce various information in the description of the intonational variants we use the following conventions:

- with bold font we marked the word with a primary accent;

- with bold and underlined letters we marked the word which carries the interrogative emphasis;
- the words with high level of energy are marked between round brackets;
- the length of accented syllables, on which a tonal event is marked, are measured including all voiced phonemes;
- the value *High* (H) of parameter HIF0 is implicit and thus, is missing in annotation of all tonal accents, which contain high level tones. It is explicit in the case of a Low value for the parameter HIF0.

In the tables from the paragraphs 3.1, 3.2 and 3.3 the information concerning the duration of accented syllables is determined by taking into account all their voiced phonemes (vowels and voiced consonants).

In the figures from the paragraphs 3.1, 3.2 and 3.3 some utterances (the wave and the F0 curve) are presented in order to exemplify different intonational variants. On the F0 curve we marked the tonal local events grouped into intonational units.

3.1 Intonational variants generated by the utterances of the oxytone text: *Ai văzut acest afiş?*

The intonational contours generated by the utterances of the oxytone text: "*Ai văzut acest afiş?*" are composed either by one, two or three intonational units.

One intonational unit and two prominent accents characterise the intonational variants from the Table 1: the first accent is on the verb and is generated by a weak tonal event characterised by a low-level topline. Instead a long duration of accented syllable makes prominent the corresponding word. The primary tonal accent is on the final word and generates the interrogative emphasis by a significant rising tone level.

During the first accent a low level limits the tone variation and thus it is not a prominent tonal event. It becomes prominent by increasing its length and the energy. The second accent is a primary tonal event H* (AT7, LS50) or L+H*(TM23, TM30).

Table 1. The "one unit" intonational variants corresponding to the utterances of the text $(Ai\ văzut\ acest\ afiş?)_A$

Intonational unit				Word with focus intention
Utterance code	Text	Tone sequence	Duration [sec.]	
AT7	**(Ai văzut)** acest **afiş**	H* (HIF0=L) $H^{(*)}$ H%	0.190 0.110	any word
LS50	**(Ai văzut)** acest **afiş**	H* (HIF0=L) $H^{(*)}$ H%	0.170 0.090	văzut
TM23	**(Ai văzut)** acest **afiş**	H* (HIF0=L) L+$H^{(*)}$ H%	0.160 0.140	văzut
TM30	**Ai văzut** acest **(afiş)**	H* (HIF0=L) L+$H^{(*)}$ H%	0.120 0.160	afiş

Figures 1 and 2 present the utterances TM23 and LS50. The last accent into utterance TM23 (Figure 1) is stronger (L+H* type) than the corresponding accent of H* type in utterance LS50 (Figure 2). In consequence, the first accent on verb is more prominent, by its duration and energy, in the case of utterances LS50 and AT7, than in the utterances TM23 and TM30.

Dividing the text into two intonational units it is justified by its syntactic structure: *verb+object* or its semantic structure: *thema* (the verb) + *rhema* (object). If both, the thema and rhema are focused, two accent units result. The most prominent accent from those, which generate the focuses, becomes the nuclear accent. In LS46 the nuclear accent is on the object, while in AT10, it is on the verb. The second primary accent is considered post or prenuclear, in respect to the nuclear accent position. The two units can be either in an intonational structure of type "A - A" or "A - B" (Table 2).

The F0 contour of the LS42 utterance is composed by a first unit within which it is accented the verb with an L+H* accent type (Figure 3). The interrogative emphasis is generated within the second unit, on the final word, afiş, by the tonal contrast of the tone sequence L+H* ˆH%. The duration of the accented syllable is of 0.107 msec and the accent becomes more prominent.

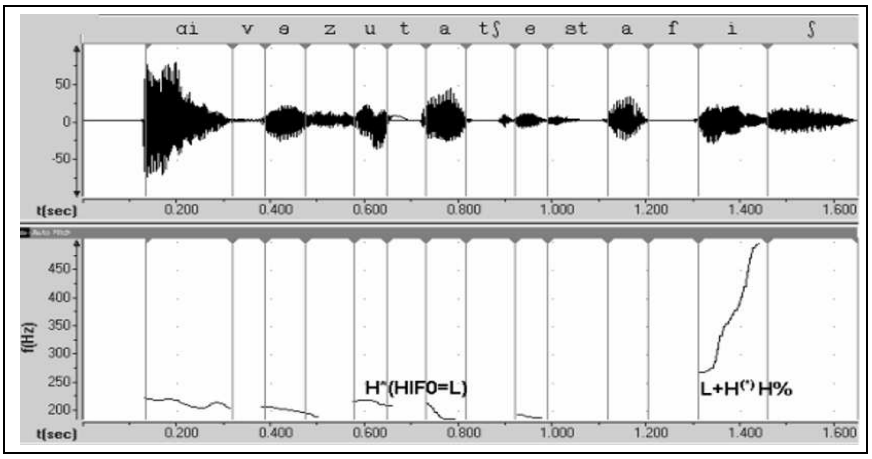


Figure 1. TM23: An utterance of the text *(Ai văzut acest afiş)*$_A$? with the melodic contour composed of one intonational unit

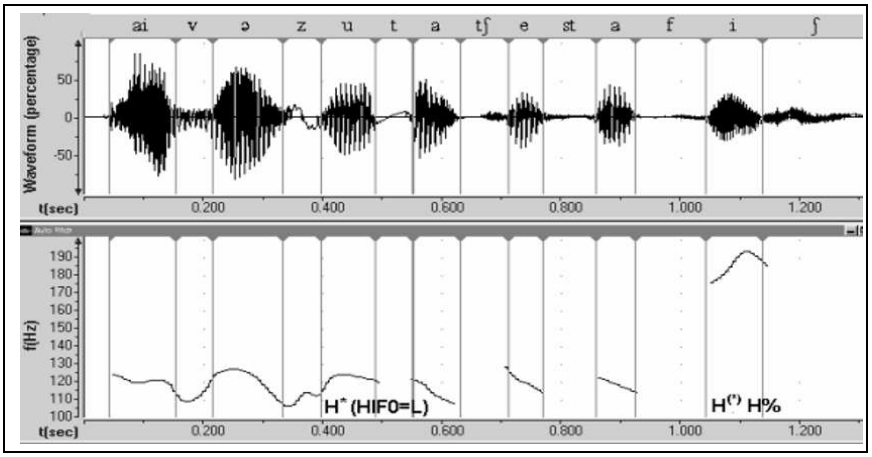


Figure 2. LS50: An utterance of the text (*Ai văzut acest afiş)*$_A$? with the melodic contour composed of one intonational unit

Table 2. The "two units" intonational variants corresponding to the utterances of the text *(Ai văzut)(acest afiş?)*

Intonational unit 1				Intonational unit 2			Word with focus intention
Utterance code	Text	Tone sequence	Duration [sec.]	Text	Tone sequence	Duration [sec.]	
LM 42	Ai văzut	L+H$^{(*)}$	0.146	**acest afiş**	^H* L$^{(*)}$L%	0.088 0.086	văzut
AT 10	**(Ai văzut)**	L+H$^{(*)}$	0.200	**acest afiş**	!H* ^H$^{(*)}$ H%	0.098	văzut
IM38, IM41	**(Ai văzut)**	L+H$^{(*)}$	0.260	**acest afiş**	^H* L$^{(*)}$ L%	0.105 0.120	văzut
LS 42	**(Ai văzut)**	L+H$^{(*)}$	0.113	acest **afiş**	H* L+H$^{(*)}$ ^H%	0.066 0.107	any word
LM 38	**Ai văzut**	L* (HIF0=L)	0.120	**acest afiş**	H* H+L$^{(*)}$ L%	0.058 0.094	any word
IM30, 34	(Ai văzut)	H* (HIF0=L)	0.100	**acest afiş**	H* H+L$^{(*)}$ L%	0.045 0.120	any word
LM 46	Ai văzut	H* (HIF0=L)	0.120	**acest afiş**	H* L$^{(*)}$ L%	0.050 0.096	afiş

In the case of LM42 utterance (Figure 4) the accent of L+H* type, on the verb, generates the most prominent contrast *low-high* and in consequence, the interrogative emphasis. The second unit of type "B" corresponds to the nominal group, with an accent of H* type on the first word and with an accent of L* type on the second.

In Table 2, the last three variants have a first intonational unit without the significant final increasing of pitch (approximately, a flat melodic segment, at a low level), corresponding to the verb *"ai văzut"*. The second unit of type "B" keeps together the words *"acest"* and *"afiş"* (Figure 5). The interrogative emphasis is generated by the tonal contrast between low average level of the first unit and the high average level within the second.

The intonational contour divided in three units is generated in the case of focusing the word "*acest*" placed in the middle of sentence. The modality to focus used by some speakers consists in isolating the word from the left one and from the right one, into a separated unit, within which it is stressed. The intonational contour with three units is generated in the case of jerky utterances, too.

In Table 3 the intonational variants divided in three intonational units are presented and Figure 6 illustrates one of them.

Table 3. The "three units" intonational variants corresponding to the utterances of the text *(Ai văzut)$_A$ (acest)$_A$ (afiş)$_A$*?

Intonational unit 1				**Intonational unit 2**			**Intonational unit 3**		
Utt. code	Text	Tone sequence	Duration [sec.]	Text	Tone sequence	Duration [sec.]	Text	Tone sequence	Duration [sec.]
LS 41	**Ai *văzut***	L+H$^{(*)}$	0.142	**acest**	H$^{(*)}$	0.078	afiş	L$^{(*)}$ H%	0.130
LM 41	**Ai văzut**	L+H$^{(*)}$	0.150	**acest**	ˆH+!H	0.055	afiş	L$^{(*)}$ H%	0.110
LS 54	**Ai văzut**	L+H$^{(*)}$	0.140	**(acest)**	L*+H H-	0.120	afiş	L$^{(*)}$ H-%	0.140
LM 50	**Ai văzut**	L+H$^{(*)}$	0.120	**(acest)**	L*+H H-	0.091	afiş	L*+H H-%	0.120
AT 14	**Ai văzut**	L+H$^{(*)}$	0.200	**(acest)**	ˆH$^{(*)}$	0.120	afiş	ˆH$^{(*)}$H%	0.120
LS 44	Ai văzut	H$^{(*)}$	0.092	**acest**	H$^{(*)}$	0.090	afiş	L$^{(*)}$ H%	0.130

In Figure 6 the LM50 utterance with semantic focus on the word "*acest*" is presented. All three tonal accents in the each unit are prominent but on this word, the length and the energy on accented syllables are higher than other accented syllables in the sentence. The interrogative emphasis (modal focus) is on the final word "*afiş*", and it is generated by a large variation of the F0 frequency between low level into the pitch accent L* and boundary tone H%.

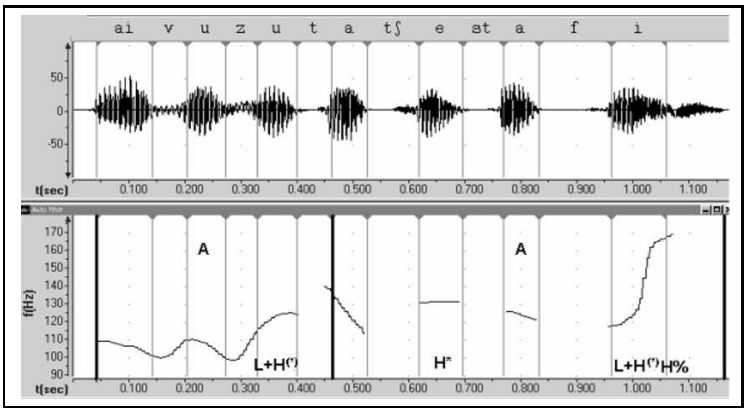


Figure 3. LS42 An utterance of the *text (Ai văzut)$_A$ (acest afiş)$_A$*? with the melodic contour composed of two intonational units

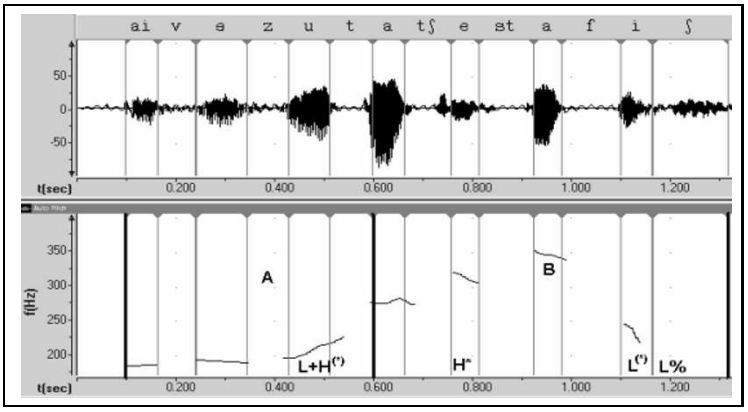


Figure 4. LM42: An utterance of the text *(Ai văzut)$_A$ (acest afiş)$_B$*? with the melodic contour composed of two intonational units

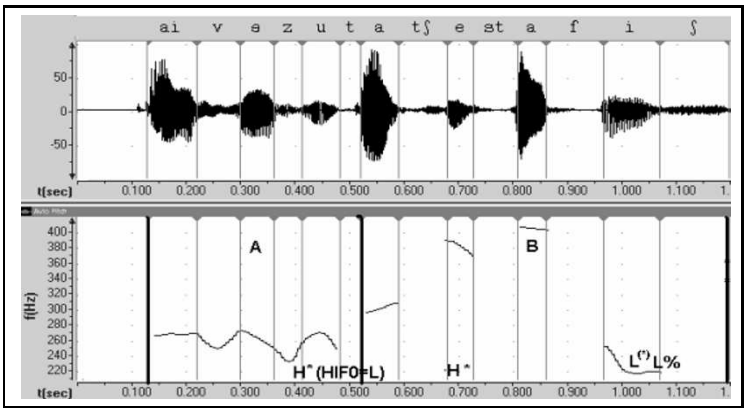


Figure 5. IM30: An utterance of the text $(Ai\ văzut)_A\ (acest\ afiş)_B$? with the melodic contour composed of two intonational units

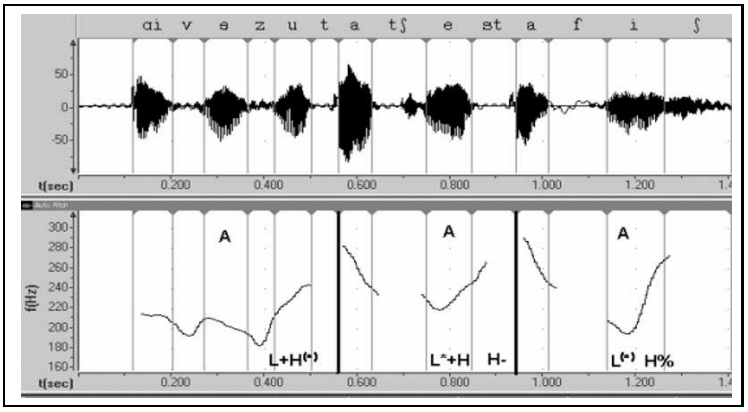


Figure 6. LM50: An utterance of the text $(Ai\ văzut)_A\ (acest\ _A\ (afiş)_A$? with melodic contour composed of three intonational units

3.2 Intonational variants generated by the utterances of the paroxytone text: *Ai văzut afişul acesta?*

The intonational variants corresponding to the text "*Ai văzut afişul acesta/ăsta?*" are presented in Tables 4-6. In Table 4 the variants with one-unit structure are presented. As in the utterance of the text "*Ai văzut acest afiş?*", there are two prominent accents: on verb (H* with HIF0=L) and on the final word (H* or L+H*). In the case of one unit structure, the final contour for this paroxytone case is ascendant-descendent, while in the oxytone case it is ascendant.

Table 4. The "one unit" intonational variants corresponding to the utterances of the text *(Ai văzut afişul acesta)*$_A$?

Intonational unit 1				Word with focus intention
Utterance code	Text	Tone sequence	Duration [sec.]	
TM3	**Ai văzut**	H* (HIF0=L)	0.170	any
	afişul		0.064	word
	acesta	L+H*L-L%	0.110	
AT1,	(**Ai văzut**)	H* (HIF0=L)	0.200	any
AT2	afişul		0.092	word
	acesta	H* !H-L%	0.120	
LS9	(**Ai văzut**)	H* (HIF0=L)	0.200	văzut
	afişul		0.059	
	acesta	H*L-%	0.075	
TM7	(**Ai văzut**)	H* (HIF0=L)	0.208	văzut
	afişul		0.606	
	acesta	H*L-L%	0.080	

In Figure 7 the F0 contour, as in all utterances of speaker TM, has a tone variation limited by a low level (200 Hz) from the beginning until the last accented syllable. During the last syllable one observe a tone variation in a large range (250 Hz) that generates the interrogative emphasis. After the last pitch accents the tone falls and returns to the low level.

In Figure 8, the intonational contour generated by a male speaker is characterised by a less prominent accent during the last accented

syllable (a pitch accent of type H*) but it generates the interrogative emphasis as in the case of speaker TM.

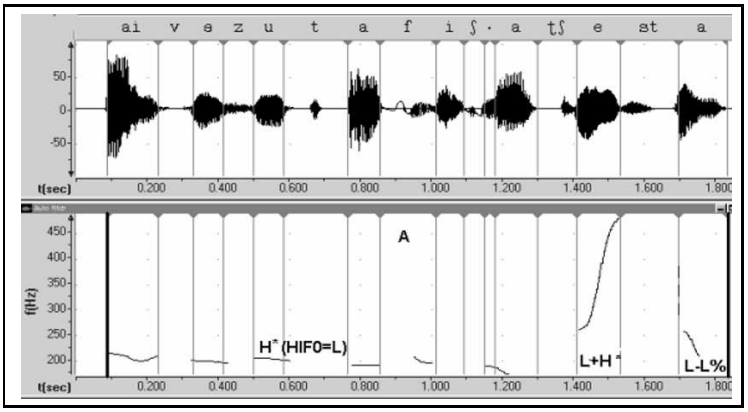


Figure 7. TM3: An utterance of the text *(Ai văzut afişul acesta)* $_A$? with melodic contour composed of one intonational unit

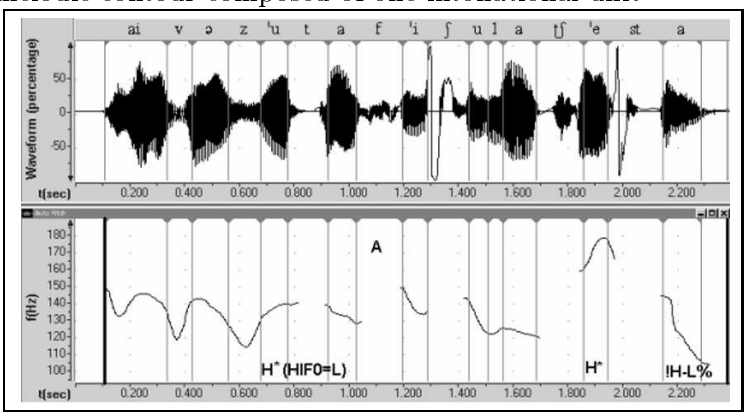


Figure 8. AT1: An utterance of the text *(Ai văzut afişul acesta)* $_A$? with melodic contour composed of one intonational unit

In Table 5 the intonations composed by two units are presented: the verb accented in the first unit and the nominal group in a second unit. The intonational structure and the position of interrogative emphasis divide them in two main types:

- one variant with the intonational structure of type "A"-"A" and the interrogative emphasis in final position, generated by the tonal contrast within the pitch accent of last word "*acesta*" (LS16);
- other variants, with the intonational structure of type "A"-"B" and the interogative emphasis in nonfinal position, generated by a broad phenomena: the tonal contrast between low average level of the first unit and the high average level within the second.

In this first case the emphasis is generated within the final word "*acesta*" based on the tonal contrast between the L* tone accent and phrase and boundary tones, H-H% (Figure 9). The final contour of the intonation is a descendent-ascendant one.

The last five variants in Table 5 represent the second type. They are characterized by a first intonational unit without the significant final increasing of pitch (approximately, a flat melodic segment, at a low level), corresponding to the verb *"ai văzut"*. The interrogative emphasis is in non-final position and is produced during the both words *"afişul acesta"*. The second type is illustrated in Figure 10 by the utterance LM3.

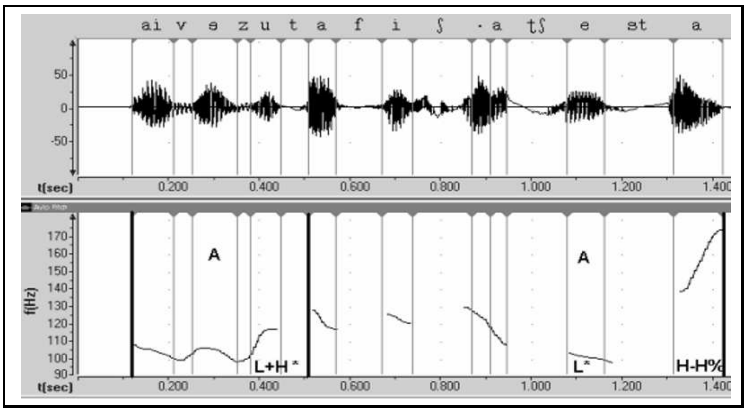


Figure 9. LS16: An utterance of the text *(Ai văzut)$_A$(afişul acesta)$_A$*? with melodic contour composed by two intonational units

Table 5. The "two units" intonational variants corresponding to the utterances of the text *(Ai văzut) (afişul acesta)?*

Intonational unit 1				**Intonational unit 2**			Word with focus intention
Utterance code	Text	Tone sequence	Duration [sec.]	Text	Tone sequence	Duration [sec.]	
LS 16	**Ai văzut**	L+H$^{(*)}$	0.110	afişul **acesta**	H* L*^H-H%	0.048 0.096	acesta
LM 3	(**Ai văzut**)	H$^{(*)}$ (HIF0=L)	0.126	afişu' **ăsta**	H* H+L$^{(*)}$L%	0.048 0.085	any word
IM 22	Ai văzut	L$^{(*)}$ (HIF0=L)	0.211	**afişul ăsta**	H* H+L$^{(*)}$!H%	0.077 0.1324	any word
LM 10	Ai văzut	L+H$^{(*)}$ (HIF0=L)	0.131	**afişu ăsta**	H* H+L$^{(*)}$L%	0.038 0.100	văzut
IM 23	(**Ai văzut**)	L+H$^{(*)}$ (HIF0=L)	0.315	**afişul ăsta**	H* H+L*H%	0.098 0.135	văzut
IM 28	**Ai văzut**	L+H$^{(*)}$ (HIF0=L)	0.190	**afişu' (ăsta)**	H* H+L*!H%	0.110 0.220	ăsta

The intonational variants in Table 6 are composed by three units generated by the fact that each word is uttered in a separate intermediate phrase with a pause between them. It is a hesitation of the speaker and it isn't an intention to focus on the word being in a middle position (Figure 11).

From a comparative analysis between oxytone and paroxytone cases of utterances of the same speaker, we conclude that speakers keep unchanged the melodic contour, except the final contour that changes like in the following:

Table 6. The "three units" intonational variants of the utterance *(Ai văzut)$_A$(afişul)$_A$ (acesta)$_A$*?

Intonational unit 1				**Intonational unit 2**			**Intonational unit 3**		
Utt. code	**Text**	Tone sequence	Duration [sec.]	**Text**	Tone sequence	Duration [sec.]	Text	Tone sequence	Duration [sec.]
LS 17	Ai văzut	H*	0.100	**(afişul)**	!H*	0.120	acesta	L*!H-H%	0.100
LS 5	**(Ai văzut)**	L+H*	0.164	afişul	!H*	0.055	acesta	L*!H-H%	0.0750

- from ascendant to ascendant-descendent one, at TM or AT speaker,
- from ascendant to descendent-ascendant at LS.

The IM and LM speaker, from south-west Transilvania region, don't change the final melodic contour.

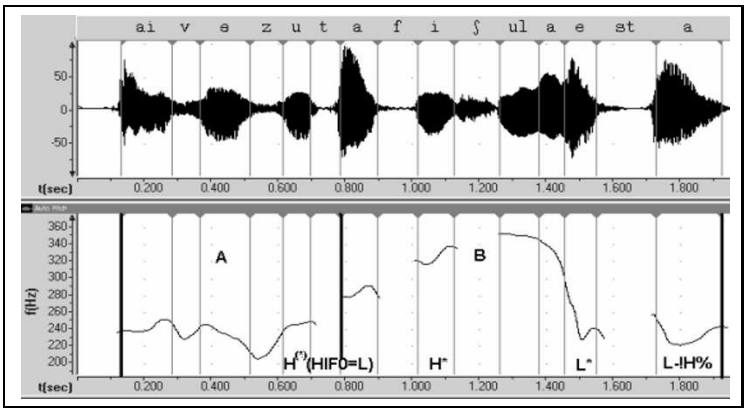


Figure 10. IM28: An utterance of the text *(Ai văzut)$_A$ (afişul ăsta)$_B$*? with melodic contour composed by two intonational units

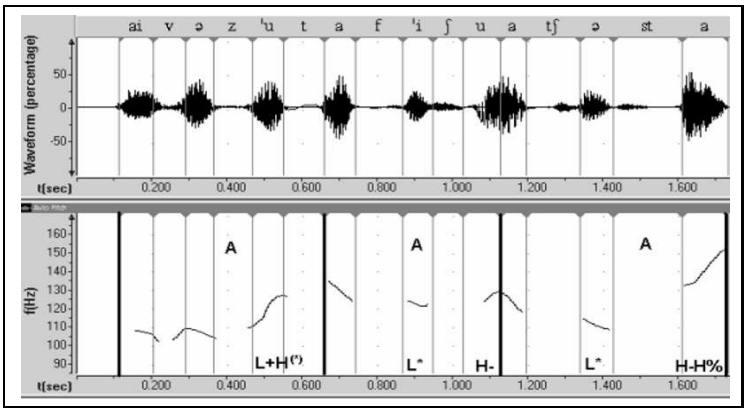


Figure 11. LS5: An utterance of the text *(Ai văzut)$_A$(afişul)$_A$ (acesta)$_A$*? with melodic contour composed of three intonational units

3.3 Intonational variants generated by the utterances of the proparoxytone text: *Ai văzut regele?*

In Table 7 and in Figure 12 one utterance of speaker LS with the intonational contour consisting of one unit is presented. The verb is uttered with a low limit tone and the emphasis is generated on the word "*regele*" beginning with the accented syllable. The highest tone of the final contour corresponds to the phrase accent H-.

Table 7. The "one unit" intonational variants corresponding to the utterances of the text *(Ai văzut regele?)$_A$*

Intonational unit 1				Word with focus intention
Utterance code	Text	Tone sequence	Duration [sec.]	
LS 23	*Ai* **văzu regele**	H* (HIF0=L) L+H* ˆH-L%	0.120 0.133	any word

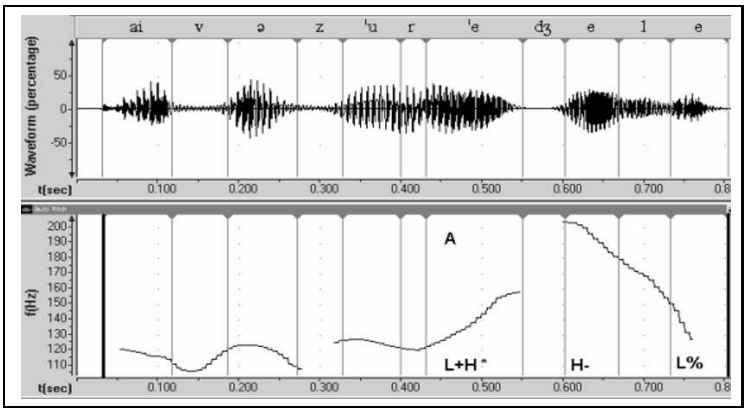


Figure 12. LS23: An utterance of the text *(Ai văzut regele?)*$_A$ with a melodic contour composed of one intonational unit

Table 8 contains several interesting variants for intonation of this proparoxyton text. From the speaker LS we extracted two intonation contours. In LS26 utterance, the intonational contour can be defined by a unit sequence "A-B" (Figure 13). Within the unit "A" a large duration and a high level of energy accent the verb. Instead the object is stressed by a prominent pitch accents of type L+H*. The final contour is ascendant-descendent and the interrogative emphasis is on the object. In LS37 utterance the verb is stressed by an increasing accent of type L+H*, while the object – by a decreasing accent of type L* (Figure 14). The interrogative emphasis is on the word "*regele*", with the low segment on the first two syllables and the high prominence on the last unaccented syllable. The final contour is descendent-ascendant.

The speaker LM generates two intonational variants presented in Table 8. Both of them have an increasing accent L+H* on a verb and the emphasis on the object *"regele"* generated in the second unit. In LM 23 utterance (Figure 15), in the second unit of type B, the word is focused with an accent of type H*+L on stressed syllable, and a tone sequence L-L% on the next unaccented syllables. The final contour is

ascendant-descendent.

Table 8. The "two units" intonational variants corresponding to the utterances of the text *(Ai văzut)(regele?)*

Intonational unit 1				**Intonational unit 2**			Word with focus intention
Utterance code	Text	Tone sequence	Duration [sec.]	Text	Tone sequence	Duration [sec.]	
LM 23	Ai **văzu'**	L+H*	0.152	**(regele)**	H* L- L%	0.172	any word
LM 22	Ai **văzu'**	L+H*	0.160	**(regele)**	H+L* H-L%	0.157	any word
LS 26	**(Ai văzut)**	H* HIF0=L	0.337	**regele**	L+^H* H-L%	0.215	văzut
LM 34	**(Ai văzu')**	L+H*	0.158	**regele**	H* L-L%	0.131	văzut
AT 32	**(Ai văzut)**	L+H$^{(*)}$	0.250	**regele**	H* L-L%	0.220	văzut
LS 37	Ai văzut	L+H$^{(*)}$	0.129	**(regele)**	L* L- H%	0.138	regele
LM 27	*Ai* ***văzut***	L+H$^{(*)}$	0.171	***(regele)***	L* H- L%	0.155	regele
AT 36	*Ai* ***văzut***	L+H$^{(*)}$	0.197	***(regele)***	L+H* H-L%	0.270	regele

In LM 22 utterance (Figure 16), in the second unit of type A, the word is focused by an accent of type L* on stressed syllable followed by a tone sequence H-L% on the next unaccented syllables. The final contour is descendent-ascendant-descendent.

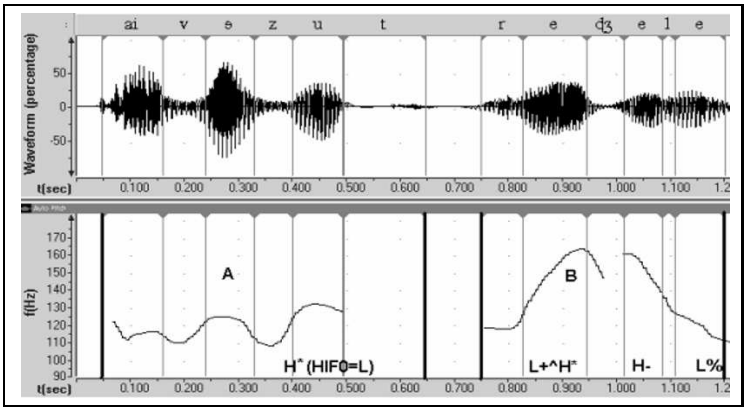


Figure 13. LS26: An utterance of the text *(Ai văzut)$_A$ (regele?)$_B$* with a melodic contour composed of two intonational units

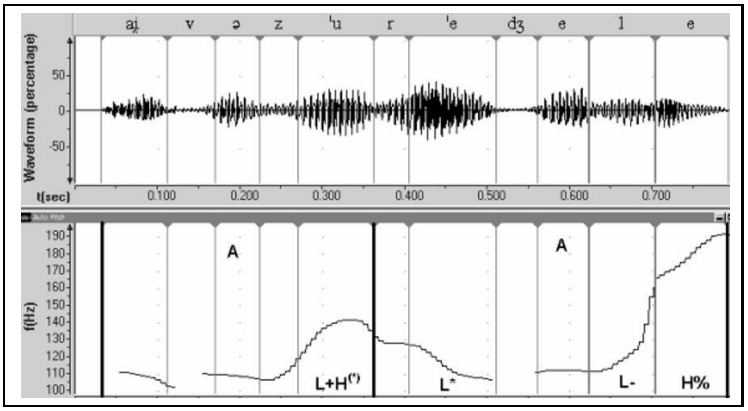


Figure 14. LS37: An utterance of the text *(Ai văzut)$_A$ (regele?)$_A$* with a melodic contour composed of two intonational units

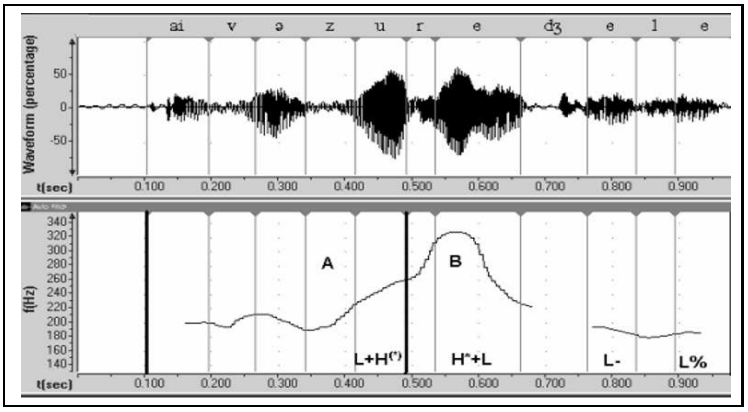


Figure 15. LM23: An utterance of the text *(Ai văzut)$_A$(regele?)$_B$* with melodic contour composed by two intonational units

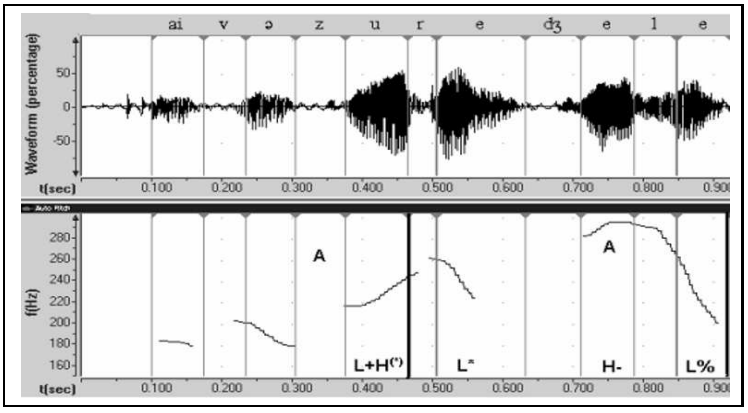


Figure 16. LM22: An utterance of the text (Ai văzut)$_A$(regele?)$_A$ with a melodic contour composed of two intonational units

4 Conclusions

From this study results that the intonation in interrogative utterances is characterised by an interrogative emphasis generated by a tonal contrast within one of the intonational units that compose the F0 contour or between the average tone of two consecutive units.

In first case the contrast implies two syllables (Figure 3) or only one accented syllable of the same word (Figure 13). In the second case the contrast between the average tone of two consecutive units is illustrated in Figure 11.

Within one intonational unit there is a prominent accent, generated by tonal variation or by duration and energy. The nuclear accent of the sentence is the most prominent accent from all units. The interrogative emphasis needs a tonal contrast in all cases. The most prominent accents that can generate an emphasis are the bitonal ones L+H* or L*+H, or H+L*. Both monotone and bitonal accents become prominent by increasing their energy and duration.

Some general rules for Yes-No questions speech synthesis, without any indication of intonation, are the following:

- Implicit accents must be provided on verb and final word. The final accent must have a more prominent tonal contrast and the verb can be accented specially by duration and energy;
- One or two intonational units can compose the intonational contour in one of intonational variants presented in this paper;
- The indications for semantic focus must drive the synthesis in increasing prominence of implicit accents or in generating others, by dividing the F0 contour in many intonational units.

The author D.R. Ladd characterised in [2] the Romanian intonation for the Yes-No questions by the sequence L* HL in one variant and H L*HL in the second. By "*" he marked the nuclear accent. He illustrates his presentation by the neutral utterances of the text "*Ai văzut afişul acesta?*" and "*Ai văzut regele?*"

The first sequence L* HL corresponds to the utterances characterised by one-unit intonational contour (Table 1, 4). The tones from D.R.Ladd's description correspond to our interpretation in following manner:

- the tone L* corresponds to the low average tone that limits the tone variations during pitch accents (H*), before the last accented syllable (in ToBI annotation system, parameter HIF0=L);
- the tone H corresponds to the increasing accent (H* or L+H*) on last accented syllable;
- the last tone L corresponds to the boundary tone L% only in the paroxytone cases.

Comparing the second sequence with the intonational variants resulted from our analysis we conclude that this corresponds to the variant composed by two intonational units: in the first, the verb is accented by a L+H* type accent and in the second, the emphasis is generated by a sequence L*H-H%. The last tone "L" from D. R. Ladd's description can be found only in proparoxytone text cases, uttered by speakers from south-west Transilvania and Banat (like as LM22 utterance, here illustrated in Figure 16).

The conclusions of Yes-No question analysis are important for linguistic studies and in Romanian speech synthesis.

References

[1] Laurenţia Dascălu-Jinga, *Melodia vorbirii în limba română*, Editura Univers enciclopedic, Bucureşti, 2001.

[2] D.R. Ladd, *Intonational Phonology*, Cambridge University Press, 1996

[3] M.Beckman, G.Ayers, *Guidelines for ToBI Labelling* (version 3, March 1997), http://www.ling.ohio-state.edu/research/phonetics/E_ToBI, 1997.

[4] V.Apopei, D.Jitca, H.N.Teodorescu, *Implementation of stress and emotion rendering rules in synthesized speech*, Trends in speech technology, Romanian Academy Publishing, pg. 67–72, 2005.

[5] V.Apopei, D.Jitca, *Romanian Intonational Annotation Based on Tone Sequence Model*, SASM 2005, Iaşi, Romania, May 5-7, 2005.

[6] A.Turculeţ, V.Apopei, D.Jitcă, *Aspecte ale intonaţiei propoziţiilor interogative totale cu structura* VO(Adj), Anuar de Lingvistică şi Istorie Literară, Iaşi.

V.Apopei, D.Jitcă, A.Turculeţ,

Received February 1, 2006

Institute for Computer Science,
Romanian Academy, Iaşi Branch
B-dul Carol I, nr. 22A, 6600 IAŞI, ROMANIA
E–mail: *vapopei@iit.tuiasi.ro*

Computer Science Journal of Moldova, vol.**14**, no.1(40), 2006

Integrity and correctness checking of a lexical database*

S. Cojocaru A. Colesnicov L. Malahova

Abstract

A Romanian lexical database being the core of the Romanian Reusable Resources for Natural Language Technology should be thoroughly checked for integrity, correctness, and completeness before to be made widely available. This case study is presented.

1 Introduction

The core of Language Engineering applied to language understanding and generation resides in the acquisition of sufficient resources in the languages to be treated, which are used to provide morpho-syntactic, lexical and semantic information, as necessary for grammar development, statistical data for language models, etc.

Aiming this, the Romanian Reusable Resources for Natural Language Technology [1, 2] (the Resources) were developed. The Resources consist of a database with the linguistic information for Romanian at the word level, and a set of service programs. Extraction of linguistic information from the database can be done by formulating SQL queries.

Depending on user's command level in Romanian, it is possible to develop different applications based on the Resources for non-Romanian speaking users, ordinary Romanian speaking users, and expert users.

Applications dedicated to the non-Romanian speaking user may include a kind of e-learning system for Romanian morphology. The DB and its existing viewers may be used by international students and

Supported by INTAS project Ref. Nr. 05-104-7633

language minorities in Moldova and Romania as directory in Romanian morphology.

Applications dedicated to the usual Romanian speaking user are Web interfaces intended to allow the use of morphological variations of Romanian words and synonyms in standard Web browsers. Web-service of spelling checking may be also developed.

Expert users of Romanian language can use the Resources to support dictionary development, including advanced lexicographic operations and support of complex browsing among dictionaries of different types. The DB with its programming possibilities seems to be a flexible and powerful tool for these operations.

Before to make our Resources widely available we should pass the stage of their correctness and integrity checking.

As the volume of the Resources is as big as hundreds thousand or even millions items, we should check their correctness and integrity in maximally automated mode. More precisely, we should use some programs to select suspicious information and to propose to the operator or the expert in philology to make the final decision.

The list of suspicious information items should be as short as possible for better reviewing by a specialist.

This article describes our approach to the checking of integrity and correctness of the lexical resources presented as a DB containing Romanian words, their morphological derivatives, synonyms, English and Russian translations, etc.

In Section 2, we shortly describe the DB structure. This section is mere technical and gives the presentation of information we work over.

Not only the information structure, but the methods the information for the DB was obtained influence the techniques of its checking. That's why we discuss methods of DB population in Section 3.

By applying automated methods, we can reveal only part of errors and suspicious items. The visual checking performed by an operator remains an important verification method. In Section 4, we describe the DB visualization tools (viewers) that were used for this.

In Section 5, we discuss some techniques we used to check the DB integrity and information correctness.

2 Database structure: main and auxiliary tables

The Resources DB has six main tables and a lot of auxiliary tables. Auxiliary tables contain different codes used in the main tables, e.g., codes of morphological characteristics or languages.

Six main tables are words, words_engl, words_rus, word_flexies, word_synonyms, word_translations. The former three tables map, correspondingly, Romanian, English, and Russian word to the numerical codes. These numerical codes are used in the latter three tables instead of textual word presentation. E.g., the word_synonyms table contains synonym pairs that consist of two numbers of the corresponding Romanian words from the words table.

There is other necessary information in these tables. Some examples follow.

The words table contains the part_code (part of speech) and field_code (domain of the word usage) fields.

Numerically encoded word/translation pairs in the word_translations table are marked by the language code to distinguish English and Russian translations.

The word_flexies table contains the flexy_word character field keeping derivatives of Romanian words. Each derivative is associated with its lemma in the words table through the integer prim_word_code field. The integer morpho_code field substantiates morphological information (tense, number, case, etc.).

As for auxiliary tables, the morpho_code field is substantiated using not one single table but ten auxiliary tables in correspondence to ten Romanian parts of speech: noun, adjective, verb, numeral, adverb, pronoun, preposition, conjunction, article, interjection. These are tables named noun_part_speech, adjective_part_speech, verb_part_speech, etc. The fields in these tables contain codes of Romanian morphologic categories corresponding to the part of speech.

You can see more detailed list of tables and their fields in Section 5, Tab. 1.

3 Database population

The DB population is one of the most important part of such project development. We took into account the request of highest quality of the DB population and decided therefore to populate the DB programmatically from textual information files.

For morphological information, we used a set of log files produced under our precedent projects [5]. Information for translations and synonyms was taken from different lexicographical sources [3].

First of all, a uniform format for data input was developed, and existing data files were transformed to this format.

Data input files for DB population consist of line groups.

For the word_flexies table, each group contains one word-lemma with all its derivatives (word-forms). Encoded morphological information is included with each word-form. Part of speech and domain of usage is included with each word-lemma.

For synonyms, each group contains a main word and its synonyms; figurative synonyms are marked.

For translations, each group contains a main word and its translations into English or Russian. Information on the domain of usage is attached to the main word.

The DB population program produces log that shows if words were inserted, shows word codes, and the result for each operation. Errors are marked and can be easily found. We also see how many words were entered and which words were not entered because they double the existing in the DB ones.

Another tool for DB population with morphological information is a semi-automatic program that generates all word-forms for a given Romanian words. The program is wizard-like and the input should be done by an expert linguist.

4 Viewers

Several viewers were programmed to check the DB visually, and to demonstrate possibilities and information contained in the DB.

We have five viewers: morphological characteristics viewer, word-forms (derivatives) viewer, synonyms viewer, English translations viewer, and Russian translations viewer.

All viewers have two common parts. The first common part is the input form. The user can ask not only for one word but he/she can use a regular expression with '?' meaning an arbitrary character and '*' meaning an arbitrary string (may be empty). The input form contains also five buttons that produce Romanian letters with diacritic for the case of absence of Romanian keyboard layout.

Fig. 1 shows the input form with the request to search morphological characteristics of a group of words.

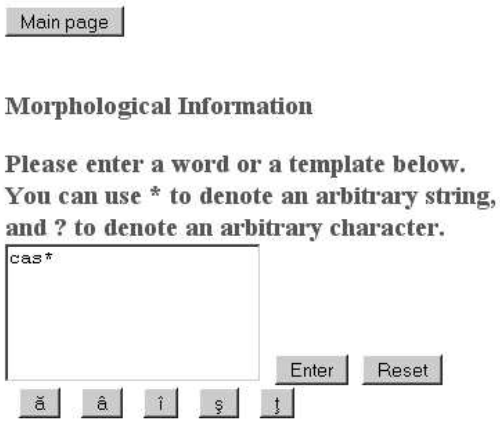


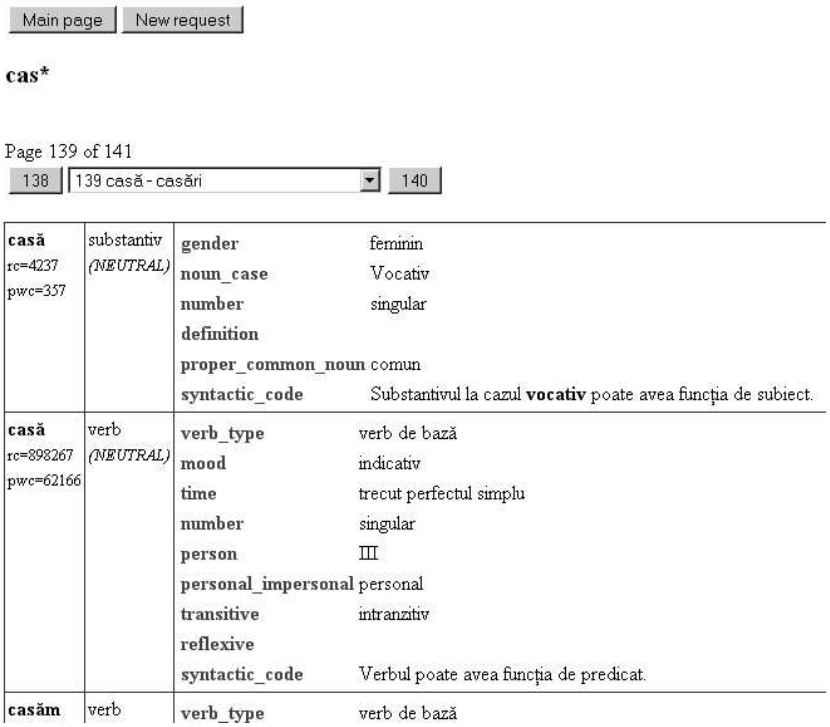
Figure 1. The input form for morphological characteristic search

The second common part of all viewers is called 'the pager' and controls the distribution of the DB data on pages (at 5 blocks per a page), page selection by its number, switching to the next or previous page, etc.

The morphological characteristics viewer queries the word_flexies table for the given word(s) and shows morphological attributes for each matching word.

The word-forms viewer queries the words table for the given word-lemma(s), then selects all derivatives of each word from the word_flexies table and shows them with their morphological attributes. The synonyms, English translations, and Russian translations viewers work analogously over the corresponding tables.

Fig. 2 shows a page produced by the morphological characteristics viewer. We asked for the regular expression 'cas*' that matches 702 derivatives in the word_flexies table.



Main page | New request

cas*

Page 139 of 141

138 | 139 casă - casări | 140

casă rc=4237 pwc=357	substantiv *(NEUTRAL)*	**gender**	feminin
		noun_case	Vocativ
		number	singular
		definition	
		proper_common_noun	comun
		syntactic_code	Substantivul la cazul **vocativ** poate avea funcția de subiect.
casă rc=898267 pwc=62166	verb *(NEUTRAL)*	**verb_type**	verb de bază
		mood	indicativ
		time	trecut perfectul simplu
		number	singular
		person	III
		personal_impersonal	personal
		transitive	intranzitiv
		reflexive	
		syntactic_code	Verbul poate avea funcția de predicat.
casăm	verb	**verb_type**	verb de bază

Figure 2. A page of found morphological characteristics

5 DB integrity

The building of a lexical resource is a difficult process. We tried to automate it maximally using specially developed programs. Meanwhile, at least three sources of errors remain that can influence the final result, namely:

- errors in the used lexicographical sources;
- errors in programs processing lexicographical information;
- operator's errors.

We can therefore suppose that each field of our DB can be potentially erroneous. We can not solve our task of information verifying using only software tools. E.g., it is impossible to decide if one word is a real synonym or translation of another word without consulting with an expert in philology. Meanwhile, it is possible to develop a set of techniques that solve this problem partially. The techniques developed for our case are described below.

5.1 Formal DB validity

First of all, we can apply formal methods to check validity of the DB structure. These methods can be formulated using the semantics and interdependencies of the DB fields and tables. All DB fields are divided for it in four categories:

1. fields containing textual representation of words (in our case, Romanian, English, and Russian);
2. fields containing references that connect different tables, e.g., numbers of Romanian words that replace words themselves in the word_synonyms table;
3. fields containing morphological and other attributes;
4. fields containing textual representation (deciphering) of attributes; these fields exist only in the auxiliary tables.

Fields of our main tables are listed in Tab. 1. For each field, its category and the method of its formal checking are shown.

Depending of the used DB engine, some formal relationships can be supported automatically.

5.2 Checking of words

Non-formal checking may be executed by variety of techniques depending on the field category. E.g., category 1 fields can be checked by usual spell checkers. For Romanian, we have our own spell checker RomSP [5]. The corresponding list of Romanian words was carefully tested and updated by developers and many users of the product, and we can take it as being quite reliable. We used also Romanian, English, and Russian spell checkers from MS Office. For Romanian, we marked words that were rejected by both spell checkers as highly suspicious. The analysis show that most of them were erroneous.

We can use other word lists, e.g., those coming with free spell checkers like ISpell [6].

A different method of word checking supposes the selection of n-grams (word fragments of n letters, $n > 2$) from the given set of words, and calculation of their frequencies. Less frequent n-grams are considered to be suspicious. Words that contain such n-grams should be checked by experts.

5.3 Checking of attributes

We saw that category 3 fields can be formally checked as containing in one of additional tables as the record number. The correspondence between fields of categories 3 and 4 can be checked informally using interval of values for different attributes but this is partial checking only. In any case, additional tables are short and can be checked visually. We can also search for unused codes in them. The correspondence of codes in the morpho_categories table and tables for each part of speech was checked by issuing requests that show in parallel decoded values of each code.

Table 1. Formal checks in the main DB tables

Table	Field	Cat.	Check method
word_flexies	prim_word_code	2	This field should be a record number in the words table.
	flexy_word	1	This field should be non-blank; only Romanian letters are permitted.
	morpho_code	3	The check is made through the chain words.prim_word_code → part_of_speech → record_number in the corresponding table.
word_synonyms	prim_word_code	2	This field should be a record number in the words table.
	synonym_code	2	This field should be a record number in the words table.
	figurat_code	3	0 or 1; shows if the synonym has figurative meaning.
word_translations	lang_code	3	Record number in the languages_table; ≠Romanian.
	prim_word_code	2	Record number in the words table.
	translation_code	2	Record number in the word_engl or word_rus depending of language.
	figurat_code	3	0 or 1.
words	prim_word_code	2	This code should be used in the word_synonyms, word_translations, or word_flexies tables.
	part_code	3	Record number in the part_code table.
	word_ii	1	Non-blank field; only Romanian letters permitted.
	word_aa	1	Non-blank field; only Romanian letters permitted.
	field_code	3	Record number in the field table.
words_engl	record_code	2	Used in translations with lang_code=English only.
	word_engl	1	Non-blank field; only English letters, spaces, and punctuation (e.g., apostrophe) permitted.
words_rus	record_code	2	Used in translations with lang_code=Russian only.
	word_rus	1	Non-blank field; only Russian letters, spaces, and punctuation permitted.

5.4 Checking of references

The next category of checks is search for duplicates. Our DB population programs query for existence of the information before its insertion into any of tables, therefore, absence of duplicates can be supposed. Meanwhile, search for duplicates can expose some errors in the prepared data for population of the DB, or in DB population programs themselves.

In the words table the unique field is prim_word_code. The corresponding information consists of the Romanian word in its textual form, its part of speech and field of usage. These data are checked for uniqueness during DB population. Non-unique combination found means something wrong with these programs, and we can check their logs visually for this combination.

We do not enter specific field of usage for a word where we enter its morphological derivatives. In this case, the corresponding field is always set to 1 ("general"). Therefore, we can check for uniqueness of the combination of a word's textual form and part of speech and analyze the corresponding fields of usage and tables where are used "non-general" words. We created the list of uninflected words that coincide with some inflected pairs of text and part of speech, and the list of "truly" uninflected words. We discovered several cases when the information for synonyms and translations was erroneous.

Moreover, we checked the words table for uniqueness of word's textual form ignoring even its part of speech. In Romanian, adjective can coincide with adverb and noun can coincide with adjective, but such cases are relatively rare. It differs from English where the same word can be verb or noun as a rule. This check permitted to detect several errors also.

Uniqueness of records in the word_flexies, word_synonyms, word_translations, word_engl, and word_rus tables is also checked during DB population. The corresponding check can be performed after population to test the DB population programs.

5.5 Statistics of derivatives

We performed also the following informal semantic checks.

Normally, Romanian words have some standard number of inflective derivatives depending of the part of speech, e.g., 35, 39, or 40 for verbs, etc. We queried for the actual number of derivatives for words from the words table. Fig. 3 shows the result of the first such test in the aggregated form. part_of_speech= 1 means 'verb', etc. You see in the graph, e.g., one verb with 160 derivatives.

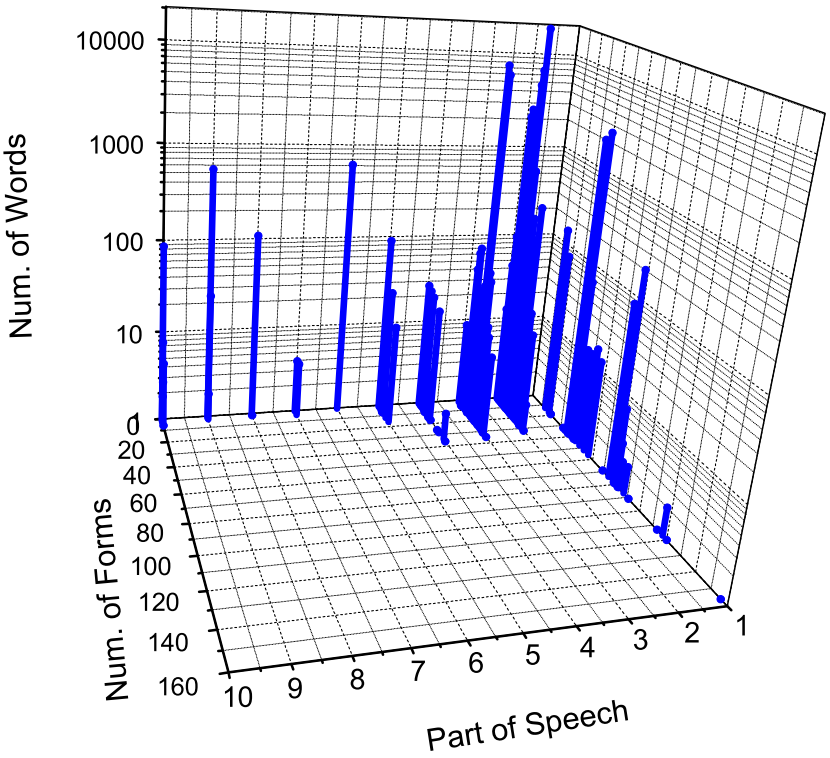


Figure 3. Statistics of derivative number

The unsuspected number of derivatives for some words permitted

us to correct some errors. E.g., it was found analyzing the case of verbs with more derivatives than necessary that some details of Romanian grammar were misunderstood during the design stage.

5.6 Checking through parallel dictionaries

Parallel dictionaries are very useful and widely used in computer linguistics (see, e.g., [4]). Our DB contains translations of many Romanian words into English and Russian. We could not get sufficient results from the English translations. The Russian translations permitted us to formulate several useful criteria because Russian is a highly inflective language like Romanian.

We used endings of Russian translations, that are more or less standard depending of part of speech, for:

- Check for words that are not verbs but Russian translations have "verbal" endings -ти -тись -ть -ться -чь -чься. We found 4119 of them, being mostly OK, but several errors were found.
- Check for words that are not adjectives but Russian translations have endings -ая -ев -ий -ин -ов -ое -ые -ый -ье -ья. No such words were found.
- Check for words that are not adverbs but Russian translations have endings -е -о -у -ем -ём -мя -ой -ом -ски. This check was not so successful (18974 words) but we shortened the result by deleting all verbs, adjectives, and nouns, and found several errors more.

5.7 DB completeness checking

The following test can be proposed to check the DB completeness. Having a list of Romanian words from any source, it is possible to sort it and to compare with the sorted list of words from the words or word_flexies tables. The word_flexies table should be used if word derivatives are permitted. This technique finds words that do not exist in our DB, and we can add them.

5.8 Correction of errors

As errors were found, they were corrected in the source data files. At a small quantity of corrections, erroneous records were deleted taking into account all interdependencies, and the corresponding part of the data file was entered anew. Having a lot of corrections, we populated anew the whole DB (six main tables) that takes quite acceptable time.

6 Conclusions

We selected the DB as linguistic information stock because of possibility of quick parallel and distant access, flexibility of possible queries, wide use and availability of the corresponding programming techniques. Other forms of information presentation like, e.g., word lists, can be easily obtained from the DB. Applications can be developed using our DB directly or indirectly.

The information containing in the DB should be thoroughly checked using different techniques. We proposed a set of methods that were found useful in our case. The discussed techniques can be applied at checking of lexical information in other cases.

Acknowledgements

The work was supported by the grant MM2-3042 of CRDF/MRDA.

References

[1] E. Boian, C. Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahova. *Lexical resources for Romanian.* In: Scientific memoirs of the Romanian Academy, ser.IV, **vol. XXVI,** Bucharest, Romania, 2005, pp. 267–278.

[2] E. Boian, C. Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahova. *Lexical Resources for Romanian – a project overview.*

In: Proceedings of Symposium on Intelligent Systems and Application, September 19–20, 2003, Iaşi, România, 12 pp. – ISBN 973–97737–2–9.

[3] E. Boian, C. Ciubotaru, S. Cojocaru, A. Colesnicov, V. Demidova, L. Malahova, T. Verlan. *Elaboration of tools to support an electronic dictionary of synonyms and transaltions.* In: International Conference "Trends in the Development of the Information and Communication Technologies in Education and Management", March 20–21, 2003, Academy of Economic Studies, Chişinău, Republic of Moldova, pp. 175–177. – In Romanian.

[4] D. Tufiş and A.M. Barbu. *Pevealing Translator's Knowledge: Statistical Methods in Constructing Practical Translation Lexicons for Language and Speech Processing.* International Journal of Speech Technology **5**, 2002, pp. 199–209.

[5] L. Malahova, A. Colesnicov. *Implementation of the Romanian Spelling Pack for Windows.* In: The International Conference on Technical Informatics CONTI'96. Proceedings. Computer Science and Engineering, **vol. 1**, Timişoara, România, 1996, pp. 23–28.

[6] `http://www.gnu.org/software/ispell/ispell.html`

S. Cojocaru, A. Colesnicov, L. Malahova,

Received March 10, 2006

Institute of Mathematics and Computer Science,
5 Academiei str.
Chişinău, MD–2028, Moldova.
E–mail: *sveta@math.md*, *kae@math.md*, *mal@math.md*