FipsRomanian: Towards a Romanian Version of the Fips Syntactic Parser

Violeta Seretan, Eric Wehrli, Luka Nerima, Gabriela Soare

LATL - Language Technology Laboratory
University of Geneva
2, Rue de Candolle, 1211 Geneva, Switzerland
Email: {violeta.seretan, eric.wehrli, luka.nerima, gabriela.soare}@unige.ch

Abstract

We describe work in progress on the development of a full syntactic parser for Romanian. This work is part of a larger project of multilingual extension of the Fips parser (Wehrli, 2007), already available for French, English, German, Spanish, Italian, and Greek, to four new languages (Romanian, Romansh, Russian and Japanese). The Romanian version was built by starting with the Fips generic parsing architecture for the Romance languages and customising the grammatical component, in close relation to the development of the lexical component. We describe this process and report on preliminary results obtained for journalistic texts.

1. Introduction

A long-term parsing project undertaken in our laboratory since the 1990s has led to the development of a large-scale, full syntactic parser, called Fips (Wehrli, 1997; Wehrli, 2007; Wehrli and Nerima, 2009). Initially available for French, Fips was later extended to English and, successively, to German, Spanish, Italian, and Greek. A number of other languages were also considered and reached different stages of development.

In the last years, sustained efforts have been made to develop a Romanian version, through the preparation of the necessary lexical resources and grammar descriptions, and their formatting so that they comply with the formalisms used by Fips. Recently, these preliminary steps have approached completion and the implementation of the grammar has started. Thus, the Romanian parser, FipsRomanian, has begun to take shape.

In this paper, we describe the current state of FipsRomanian and present its first results. The paper is organised as follows. In Section 2, we review the existing work aimed at building parsers for Romanian. In Section 3, we introduce the Fips multilingual parser, by specifying the precise nature of the analysis provided and briefly presenting its general parsing algorithm. In Section 4 we discuss the preparation of the resources needed for implementing the Romanian version, as well as the language-specific issues encountered. Parsing results are presented in Section 5, followed by final remarks in Section 6.

2. Related work

Romanian is a relatively privileged language, insofar as lexical resources and morphological tools are concerned. Lexical thesauri (like wordnets and framenets), word aligners, (parallel) annotated corpora, POS taggers, and automated processing architectures are already available and they are successfully used in various NLP tasks, e.g., word sense disambiguation, computational lexicography, question answering, anaphora resolution, textual entailment. Cristea (2009) gives a brief review of the variety of resources and tools that exist for Romanian. A more complete picture of the NLP work on Romanian can be found in the proceedings of the workshop series *ConsILR - Consortium for the Romanian Language: Resources & Tools*, e.g., (Tradabăţ et al., 2008).[1]

This situation stands in contrast with that of syntactic tools. Little or no reference exists on work on the syntactic level, be it concerned with symbolic or statistical, shallow or deep parsing. Some steps have been taken in this direction through the creation of a Romanian dependency treebank[2] and the development of a stochastic dependency parser[3] based on it (Călăcean and Nivre, 2009).

This work (Călăcean and Nivre, 2009) appears to be the most significant in the area of syntactic parsing for Romanian. The parser learns dependency relations in a data-driven way from the treebank and is based on the language-independent, freely available MaltParser (Nivre et al., 2007). It achieves a high level of accuracy, comparable to that achieved by MaltParser for other languages, like English, Italian and Catalan (Călăcean and Nivre, 2009).

Unfortunately, there are some severe limitations imposed by the treebank used. The latter is relatively limited both in terms of size and of complexity of the syntactic structures covered (for instance, it does not contain subordinate clauses, and the average sentence length is only 9 words). Moreover, since the parser was evaluated on the same treebank, it is highly questionable whether it can successfully be applied to unrestricted text.

Another report on parsing for Romanian is given in (Şaupe et al., 2009). This work was carried out in the framework of a project on sentence analysis for text-to-speech purposes. In spite of all appearances, it is not on syntactic parsing proper, as it is not concerned with sentence structure. It is limited to the lexical level, and provides a preliminary shallow lexical analysis aimed at identifying paragraph, sentence and word boundaries.

[1] `http://consilr.info.uaic.ro/`; accessed March, 2010.

[2] Developed in the framework of the BALRIC-LING project, Balkan Regional Information Centers for HLT (`http://www.larflast.bas.bg/balric/`; accessed March, 2010).

[3] The dependency parsing can be seen as an intermediate form of parsing, between shallow and deep parsing.

To summarize, there is currently no large-scale syntactic parser available for Romanian. In this paper, we present the first steps towards building a symbolic Romanian parser able to fully parse unrestricted text. Unlike the parser of (Călăcean and Nivre, 2009), which outputs dependency relations, our parser aims to create a complete syntactic structure for the input sentence. When this is not possible, the system returns at least partial parses for the chunks in the initial sentence it succeeded to analyse.

3. Fips and its generic parsing architecture

The Romanian parser described in this paper is part of Fips, a multilingual symbolic parser which relies on generative grammar concepts (Chomsky, 1995). Given an input sentence, Fips returns a rich structural representation for it, which includes:

1. the constituent structure: a parse tree reflecting the hierarchical organisation of the words in the sentence, similarly to the c-structure in LFG, the Lexical Functional Grammar (Bresnan, 2001);
2. the interpretation of constituents in terms of arguments: a predicate-argument table identifying the grammatical relations between the main constituents of the sentence (similarly to the f-structure in LFG);
3. the interpretation of elements like clitics, relative and interrogative pronouns in terms of intra-sentential antecedents;
4. co-indexation chains linking extraposed elements (e.g., fronted NPs and *wh*-elements) to their canonical position.[4]

The constituent structure in Fips is a simplified X-bar structure containing no intermediate levels. The lexical head X of a phrase XP is directly attached under the phrase node, as are its left and right sub-constituents (see Figure 1). X stands for the usual lexical and functional categories, N (noun), V (verb), A (adjective), Adv (adverb), D (determiner), P (preposition), C (complementizer), Interj (interjection), T (tensed verb, head of a sentence), and F (predicative objects).

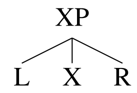

Figure 1: Structure of a constituent in Fips: X - lexical head, L - list of left sub-constituents, R - list of right sub-constituents.

Fips can be characterised a bottom-up, left-to-right parser. The parsing algorithm proceeds by iteratively performing one of the following three types of operations: creation of constituent structures corresponding to the lexical entries (Project), combination of adjacent constituents into larger constituents (Merge), and movement of constituents from the canonical position to the surface position (Move).

The application of these operations is constrained by both language-independent grammar rules (which constitute the core parser engine) and language-specific rules (defined for each language supported by the parser). Alternatives are pursued in parallel and pruning heuristics are used for keeping the algorithm tractable.

Fips can also be defined a strong lexicalist parser. In fact, one of its key components is its manually-built lexicon. The lexical entries contain rich information that guides the parser. For instance, subcategorisation information, selectional properties, collocational information and other syntactico-semantic features concur to inform the actions of the parser. From a technical point of view, Fips is implemented in Component Pascal in the object-oriented paradigm, which is suitable for modelling the generality of the core parsing engine and, at the same time, its specializations for each of the supported languages.

The following is a sample output, showing the constituent structure returned by Fips for the French sentence *Les jeux sont faits* (Figure 2). The movement of the phrase *les jeux* from the cannonical direct object position to the surface subject position, due to passivisation, is marked by the shared index *i*. The empty position e_i contains the trace left by this movement. Thanks to the co-indexation, it is possible to retrieve, in the predicate-argument table, the phrase *les jeux* as the "deep" direct object.

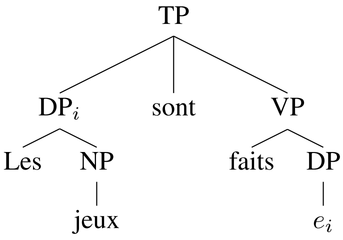

Figure 2: Sample parse tree produced by Fips for a passive sentence in French.

4. Resource development for FipsRomanian

Given the general parsing architecture described in Section 3, customizing Fips for a new language means, in principle, developing only the language-specific part of the parser. This is true for Romanian, which belongs to the Romance family of languages already represented in Fips by French, Italian, and Spanish.[5]

The language-specific part of Fips for a language L consists, on the one hand, of language-specific grammar rules for L, and, on the other hand, of the lexicon of that language L. The grammar rules specify under which condition the parser's main operations, Project, Merge and Move (cf. Section 3), can be applied in order to enable the creation of a parse tree for the input sentence. The lexicon of the

[4] In the canonical form of a sentence, the head word and the dependent word are in the typical order (e.g., the subject before the verb and the object after the verb in a SVO language). According to generative grammar stipulations, a word can move from an initial canonical position to the final surface position.

[5] A less related language could require reconsidering the core parser engine, even if this is supposed to be language-universal.

A+N		*adjectif prénominal*	marilor realizări
	a.HasFeat(prenominal)		great$_{Gen/Dat}$ achievements
	a.AgreeWith(b, gender, number)		of/to the great achievements

Table 1: Example of a grammar rule expressed in the Fips pseudo-formalism.

language contains entries for simple lexemes and complex lexemes (i.e., compound words, collocations and idioms), enriched with morphosyntactic and semantic information, whose role is to guide the parser. In this section we describe both the grammatical and the lexical component of the Romanian parser.

4.1. Grammar

The grammar specification is given in a pseudo-formalism specific to Fips, which is easy to adopt by linguists and, at the same time, close enough to the source code of the parsing program. Most rules in the specification refer to the conditions under which two adjacent constituents can be joined by the Merge operation to yield a larger constituent. A simple example is provided in Table 1. This rule enables the attachment of a prenominal adjective phrase (denoted by *a*) as a left sub-constituent of a nominal phrase (denoted by *b*), provided that the required agreement conditions are satisfied.

A most frequent type of attachment in Romanian is, however, the right attachment, where a parse tree node is inserted as a right sub-constituent of another parse tree node. In the current Romanian grammar description, there are about 100 grammar rules, a quarter of which concerning left attachments and the others right attachments.

Syntactic processes (e.g., passivization, relativisation, interrogation) are dealt with in the grammatical component of Fips that models the Romance family of languages, to which Romanian belongs. Some refinements are however needed, in particular in the account of clitics and of *wh*-elements, since they exhibit peculiar properties in Romanian with respect to other Romance languages (Monachesi, 2000; Soare, 2005; Soare, 2009).

As Monachesi (2000) shows, the Romanian clitic system is richer and involves pronouns, negation, auxiliaries as well as a restricted subclass of monosyllabic adverbs (*mai* 'again', *cam* 'little', *prea* 'too', *şi* 'also', *tot* 'still'). Their order is rigid: negation is the leftmost element, preceding the auxiliary (if any); dative clitics precede accusative clitics; and the monosyllabic adverbs fill a position between the auxiliary and the participial verb.

Romanian is similar to Spanish but differs from French and Italian in that the Accusative DPs, marked by the preposition *pe*, must be clitic doubled (cf. Example 1). Clitic doubling is optional for full Dative DPs, as shown in Example 2.

(1) a. L-am văzut pe Ion.
cl.$_{\text{Acc}}$-have seen PE Ion.
'I have seen Ion.'
b. * Am văzut pe Ion.

(2) a. (Le)-a dat fetelor nişte flori.
cl.$_{\text{Dat}}$-has given girls$_{\text{Dat}}$ some flowers.
'He/she gave the girls some flowers.'

Unlike in French and Italian, no material can intervene between the auxiliary and the participial verb, except for the above-mentioned clitic adverbs. Our implementation is consistent with Monachesi's theoretical account, which postulates a compound structure for auxiliary verbs rather than a flat structure, suitable for other Romance languages (Monachesi, 2000).

Insofar as *wh*-elements are concerned, Romanian also differs from all the other Romance languages in exhibiting multiple *wh*-movement. The *wh*-phrases are rigidly ordered, obey the Superiority Condition,[6] and no material can intervene between them (cf. Example 3).

(3) a. Cui ce a adus Moş Crăciun?
Who$_{\text{Dat}}$ what has brought Santa Claus?
'What did Santa Claus bring to whom?'
b. * Ce cui a adus Moş Crăciun?

In the current state of the development of the Romanian parser, more than half of the grammar rules for which a formal description was provided are implemented and tested. The implementation of specific movement rules (including the creation of co-indexation chains for extraposed elements) is still at an initial stage. As for the predicate-argument structure, this is specified in the lexical entries for verbs, nouns, and adjectives. The implementation of the grammar goes hand in hand with the development of the lexicon, summarized below.

4.2. Lexicon

Romanian has a rich morphology, as it inherits, in part, the Latin declension system. The relatively numerous inflected forms corresponding to nominal, adjectival and verbal lexemes[7] are obtained through morphological generation according to a given declension paradigm. The Fips morphological component allows the generation rules to be stated in a specific format, which can be read and interpreted by the system so that a paradigm is automatically obtained from its formal description.

Figure 3 shows the example of a simple inflection rule for Romanian, which is used to generate the past participle of verbs of a given inflection class—in this case, 1—by appending the suffix *at* to the radical obtained from the base represented by the present infinitive. The rule specifies that this procedure only applies to the combination masculine/neuter gender and singular number; for other

[6] Broadly speaking, the Nominative precedes the Dative, and the Dative precedes the Accusative.

[7] There are 6-7 forms for a noun (depending on the gender), about 15 forms for an adjective, and about 35 forms for a verb.

INFL
- ”at” = (cat:V, inflClass:1, base:1, tense:pastPart, gender:{masc,neut}, pers:{1, 2, 3}, num:sing).

Figure 3: Example of an inflection rule expressed in the Fips formalism.

gender-number combinations, different forms will be produced, according to the appropriate rules. This rule will generate, for instance, the past participle *curăţat* ’cleaned’ from the base *(a) curăţa* ’(to) clean’, once it is stated that *(a) curăţa* is a verb of the first inflection class.

The process of compiling the FipsRomanian lexicon went through several stages. A list of base word forms was first obtained from the DEX dictionary (DEX, 1998). An inflection class number was automatically assigned to most nouns and adjectives based on the word suffix. This allowed, in conjunction with the corresponding inflection rules defined, for the automatic generation of inflected forms for these lexemes. A part of the remaining nouns and adjectives were manually entered into the lexicon, as were pronouns, determiners and the most common verbs.

Verbs, in particular, require detailed specific information about subcategorization, selectional features, the grammatical function of arguments, the thematic function and other information (for instance, on aspect), which can only be specified manually. Nonetheless, the morphological generation process is still useful, since verb paradigms contain very numerous forms, and it is too onerous to add them manually.

Most prepositions and conjunctions were also added manually to the lexicon, whereas adverbs and interjections were basically added automatically. Our lexicon also contains proper nouns, mostly for places (cities, countries, rivers, mountains) and persons (the most usual first names and surnames, separately). Most of these were gathered from different websites.[8] Table 2 displays the composition of the current Romanian lexicon by lexical category.

Category	Forms	Lexemes
Noun (common)	254582	38655
Noun (proper)	9211	9199
Pronoun	157	74
Clitic	22	5
Adjective	76341	14296
Verb	49249	3604
Adverb	842	842
Interjection	412	412
Preposition	158	75
Conjunction	73	56
Determiner	45	45
Total	391092	67263

Table 2: Composition of the Romanian lexicon.

In addition to single-word entries, the Fips lexicon also contains multi-word entries. A first category of multi-word entries is represented by compound words (for instance, complex prepositions, conjunctions and adverbs: *de jur împrejurul* ’around’, *dat fiind că* ’given that’, *până când* ’until’, as well as some proper nouns: *Câmpulung Moldovenesc*). These have, however, the status of single lexemes, since they behave like simple words.

A second category is represented by collocations, which cover, as a special case, the idioms. Collocations are language-specific restricted combinations of words, like *a atrage atenţia*, ’to draw attention’ (lit., *to attract atention*). Like idioms, collocations pose production problems to non-native language speakers, but unlike idioms, they do not really pose comprehension problems. Idioms (e.g., *a pune la punct* ’to fix’, lit. *to put to point*) are the semantically uncompositional extreme of the collocations continuum (McKeown and Radev, 2000). On the other extreme, one finds collocations that are more similar to free combinations, like *mare importanţă* ’high importance’ (lit., *big importance*).

Since collocations allow the insertion of additional material between the component items, they cannot be stored in the lexicon in the same way as compounds. They are stored as binary associations of lexemes, where each item can be either a single word or a multi-word existing entry (in particular, a compound or another collocation).[9] Our recent experiments of collocation extraction from Romanian corpora, based on syntactically informed methods (Seretan, 2008) and made possible by the availability of the parser, allowed us to the put the basis of the Romanian collocation lexicon, which currently contains about 600 collocations.

5. Preliminary results

Although in an incipient stage (see Section 4), FipsRomanian is already operational and can be robustly used to process unrestricted text. No preprocessing is required and there is no limitation on the sentence length or type. Example 4b presents the output obtained for the simple Romanian sentence in 4a. The associated parse tree is depicted in Figure 4.

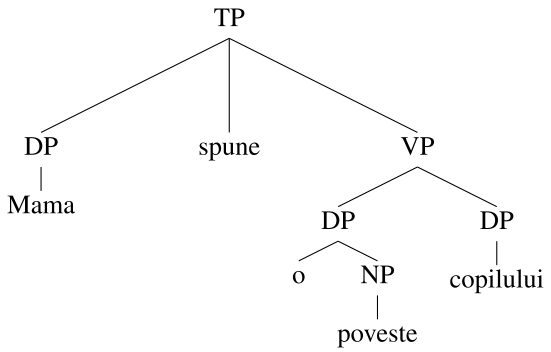

Figure 4: Sample analysis provided by FipsRomanian.

[8] `http://ro.wikipedia.org/` and `http://www.archeus.ro/` (accessed October, 2009).

[9] See (Nerima et al., 2010) for more information on our recursive treatment of collocations.

(4) a. Mama spune o poveste copilului.
Mother-the tells a story child$_{\text{Dat}}$
'The mother tells the child a story.'

b. [$_{\text{TP}}$ [$_{\text{DP}}$ mama] spune [$_{\text{VP}}$ [$_{\text{DP}}$ o [$_{\text{NP}}$ poveste]] [$_{\text{DP}}$ copilului]]]

Figure 5 presents the (simplified) POS-tagging output of FipsRomanian for this sentence. For each token, it presents the morphological analysis, the base form, and (if applicable) the grammatical function. For predicates—here, *spune*—the argument "table" is also displayed. In our case, this contains a subject (SUB), a direct object (DO) and an indirect object (IO).

```
Mama        NOM-COM-SIN-FEM-NOM-ACC
            mama SUBJ
spune       VER-INF-PRE-3-SIN          spune
            SUB:mama DO:poveste IO:copilului
o           DET-IND-SIN-FEM-NOM-ACC        o
            OBJ
poveste     NOM-COM-SIN-FEM-NOM-ACC
            poveste
copilului   NOM-COM-SIN-MAS-DAT-GEN
            copil IND-OBJ
.           PONC-point .
```

Figure 5: Sample POS-tagging output of FipsRomanian (simplified).

In addition to simple sentences, FipsRomanian is also able to (partly) deal with more complex sentences, which might involve grammatical processes such as passivisation, relativization, and clause subordination. An example is given in 5, concerning a relativization. The parser succeeds in identifying the noun *Clădirea* as the argument of the predicate in the relative clause, since it is able to co-index the empty argument position, the relative *care* 'which', and the noun appearing as the subject of the matrix clause (see also the explanations in Section 3).

(5) Clădirea Ministerului palestinian de Externe, care fusese avariată în raiduri anterioare, a fost distrusă complet (...)
The building of the Palestinian Ministry of Foreign Affairs, which had been damaged by previous raids, has been completely destroyed (...)

FipsRomanian was run on a body of online newspaper articles totalling slightly more than 1 million words. The average sentence length is 26.9 tokens, including punctuation. Here, we report on the grammatical and lexical coverage obtained on these data; for the moment, it is premature to provide more detailed performance results.

In our experiment, 16.2% of the total 44483 sentences could be fully parsed (i.e., the parser returned a single complete parse tree). This is a high number, given the relatively long sentence length. For the remaining sentences, only partial parse trees were returned. The average length, in tokens, for the partial structures is 5.3. In Figure 6 we give the example of the output returned by FipsRomanian for a sentence that could not be fully parsed. This sentence (shown in Example 6) was randomly chosen among the sentences in our data.

(6) Fosta multiplă campioană olimpică şi mondială la canotaj a fost prima dintre vedetele din lumea sportului care a acceptat fără rezerve invitaţia noastră de a se supune testului cu detectorul de adevăr.
The former multiple world and olympic canoeing champion was the first of the sport stars to accept without hesitation our invitation to undergo a lie detector test.

```
*** no analysis [NP[AP Fosta ] multiplă ]

[TP[NP campioană [AP olimpică ][AP[AdvP şi ]
mondială ]][PP la [NP canotaj ]] a [VP fost [DP prima
]]]

[TP[PP dintre [DP vedetele ]][PP din [DP lumea [DP
sportului ]]][DP care [DP a ]] acceptat [VP [AdvP [PP
fără [NP rezerve ]]][DP invitaţia [DP noastră [CP de
[TP[PP a [NP se ]][DP ] supune [VP [DP testului [PP
cu [NP detectorul [PP de [NP adevăr ]]]]]]]]]]]
```

Figure 6: Partial analysis output for a randomly-chosen sentence.

The lexical coverage of FipsRomanian on these data is 93.5%; 6.5% of the tokens in the corpus (77791 tokens corresponding to 19348 types) are not yet covered by the parser's lexicon. Most of these are proper nouns (39.2%). This result indicates that the lexical component of FipsRomanian is satisfactorily developed.

As mentioned in Section 4, the parsing results were used in the task of collocation extraction, by applying the syntactically informed procedure described in (Seretan, 2008). Among the top 2000 syntactic co-occurrences produced in the order of association strength as given by the log-likelihood ratio measure, a number of 606 candidates have been retained as lexicographically interesting. The extraction precision is 30.3%. In contrast, the precision obtained with the same procedure for other languages supported by Fips is around 50%. This result suggests that, even if the Romanian version of the parser is comparatively much less developed, its results are nonetheless useful for practical applications.

6. Conclusion

In this paper, we introduced FipsRomanian, the Romanian version of a multilingual symbolic parser. The development is still in progress; yet, the parser could already be applied on unrestricted journalistic texts. A thorough evaluation still has to be done. However, preliminary experiments indicate that the parsing results, even if most of them concern partial rather than complete sentence analyses, are useful for other applications.

Acknowledgements

This work has been supported by the Swiss National Science Foundation (grant no. 100012-117944). The authors gratefully acknowledge the contribution of many former collaborators to this project, including Gianina Aonofriesei, Maria Husarciuc, Diana Tradabăţ.

7. References

Joan Bresnan. 2001. *Lexical Functional Syntax*. Blackwell, Oxford.

Noam Chomsky. 1995. *The Minimalist Program*. MIT Press, Cambridge, Mass.

Dan Cristea. 2009. Romanian language resources and tools. `http://www.clarin.eu/files/cnl04_web.pdf`. Accessed October, 2009.

Mihaela Călăcean and Joakim Nivre. 2009. A data-driven dependency parser for romanian. In *Proceedings of the 7th International Workshop on Treebanks and Linguistic Theories (TLT 7)*, pages 65–76, Groningen, Holland.

1998. *DEX - Dicţionarul explicativ al limbii române*. Academia Română, Bucharest.

Kathleen R. McKeown and Dragomir R. Radev. 2000. Collocations. In Robert Dale, Hermann Moisl, and Harold Somers, editors, *A Handbook of Natural Language Processing*, pages 507–523. Marcel Dekker, New York, U.S.A.

Paola Monachesi. 2000. Clitic placement in the Romanian verbal complex. In B. Gerlach and J. Grijzenhout, editors, *Clitics in Phonology, Morphology and Syntax*. John Benjamins, Amsterdam.

Luka Nerima, Eric Wehrli, and Violeta Seretan. 2010. A recursive treatment of collocations. In *Proceedings of The seventh international conference on Language Resources and Evaluation (LREC 2010)*, Valletta, Malta.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13:95135.

Violeta Seretan. 2008. *Collocation extraction based on syntactic parsing*. Ph.D. thesis, University of Geneva.

Gabriela Soare. 2005. Romanian syntax. Technical report, Language Technology Laboratory, University of Geneva.

Gabriela Soare. 2009. *The Syntax-Information Structure Interface: A Comparative View from Romanian*. Ph.D. thesis, University of Geneva.

Andrei Şaupe, Lucian R. Teodorescu, Mihai A. Ordean, Răzvan Boldizsar, Mihaela Ordean, and Gheorghe C. Silaghi. 2009. Efficient parsing of Romanian language for text-to-speech purposes. In V. Matoušek and P. Mautner, editors, *Text, Speech and Dialogue 2009*, pages 323–330. Springer-Verlag, Berlin/Heidelberg.

Diana M. Tradabăţ, Dan Cristea, and Dan Tufiş, editors. 2008. *Lucrările atelierului Resurse lingvistice şi instrumente pentru prelucrarea limbii române*. Editura Universităţii "Alexandru Ioan Cuza", Iaşi. In Romanian.

Eric Wehrli and Luka Nerima. 2009. L'analyseur syntaxique Fips. In *Proceedings of the IWPT 2009 ATALA Workshop: What French parsing systems?*, Paris, France.

Eric Wehrli. 1997. *L'analyse syntaxique des langues naturelles: Problèmes et méthodes*. Masson, Paris.

Eric Wehrli. 2007. Fips, a "deep" linguistic multilingual parser. In *ACL 2007 Workshop on Deep Linguistic Processing*, pages 120–127, Prague, Czech Republic.