A Lexicalized Tree-Adjoining Grammar for Vietnamese

LE H. Phuong*, **NGUYEN T. M. Huyen ***, **ROMARY Laurent†**, **ROUSSANALY Azim†**

*Hanoi University of Science
Hanoi, Vietnam
{phuonglh, huyenntm}@vnu.edu.vn
†Laboratoire Lorrain de Recherche en Informatique et ses Applications
Nancy, France
{romary, roussanaly}@loria.fr

Abstract

In this paper, we present the first sizable grammar built for Vietnamese using LTAG, developed over the past two years, named vnLTAG. This grammar aims at modelling written language and is general enough to be both application- and domain-independent. It can be used for the morpho-syntactic tagging and syntactic parsing of Vietnamese texts, as well as text generation. We then present a robust parsing scheme using vnLTAG and a parser for the grammar. We finish with an evaluation using a test suite.

1. Introduction

As far as electronic syntactic resources go, a distinction can be drawn between program-dependent and reusable grammars. The formalisms of unification-based grammar have been used to develop reusable broad-coverage grammars for English, French, German, Chinese, Japanese, Korean, *etc*. However, such a grammar does not exist for Vietnamese, a language spoken by about 85 millions people around the world.

Our objective is to build linguistic resources for the task of parsing and grammar evaluation. For the parsing, we choose the LTAG (Lexicalized Tree-Adjoining Grammar) formalism to model the Vietnamese grammar. In parallel with the grammar construction, we try to build a test suite, inspired by TSNLP principles (Balkan et al., 1994), independent from linguistic theories, so that it can be used to evaluate any grammar. The test suite contains minimal phrases in a very simple form, accompanied by agrammatical derivations obtained through some linguistic test operations: inside element change, addition, deletion, and permutation.

We begin in Section 2 by briefly discussing LTAG, a powerful formalism that allows the modelling of various syntactic phenomena of natural languages. In Section 3, we present the first sizable grammar built for Vietnamese using LTAG, developed over the past two years, named vnLTAG. This grammar aims at modelling written language and is general enough to be both application- and domain-independent. Section 4 presents a robust parsing scheme using vnLTAG and a parser for the grammar which is based on LLP2, a syntactic parser developed at LORIA[1]. Finally, Section 5 discusses about our future work.

2. Lexicalized Tree-Adjoining Grammars

Tree-adjoining grammar (TAG) is a tree-rewriting formalism originally defined by (Joshi et al., 1975). The first study of this system, from the point of view of its formal properties and linguistic applicability, was carried out by (Joshi, 1985). TAGs have been used to provide linguistic analyses; a detailed study of the linguistic relevance was done by Kroch and Joshi (Kroch and Joshi, 1985).

A TAG consists of a finite set of elementary trees. The nodes of these trees are labeled with nonterminals and terminals. Starting from the elementary trees, larger trees are derived using composition operations of substitution and adjunction. In the case of an adjunction, the tree being adjoined has exactly one leaf node that is marked as the foot node (marked with an asterisk). Such a tree is called an *auxiliary tree*. Elementary trees that are not auxiliary trees are called *initial trees*. Each derivation starts with an initial tree. In the final derived tree, all leaves must have terminal labels.

Figures 1 and 2 show a sample TAG derivation with a substitution and an adjunction. Here, the three elementary trees for *laughs*, *John*, and *always* are combined: Starting from the elementary tree for *laughs*, the tree for *John* is substituted for the noun phrase (NP) leaf and the tree for *always* is adjoined at the verb phrase (VP) node.

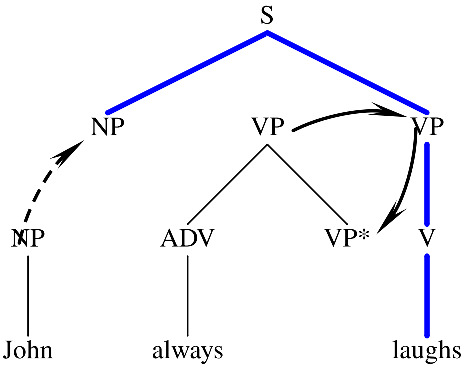


Figure 1: TAG derivation for *John always laughs.*

TAG derivations are represented by derivation trees that record the history of how the elementary trees are put together. A derivation tree is the result of carrying out substitutions and adjunctions. Each edge in the derivation tree stands for an adjunction or a substitution.

[1] Laboratoire Lorrain de Recherche en Informatique et ses Applications, http://www.loria.fr

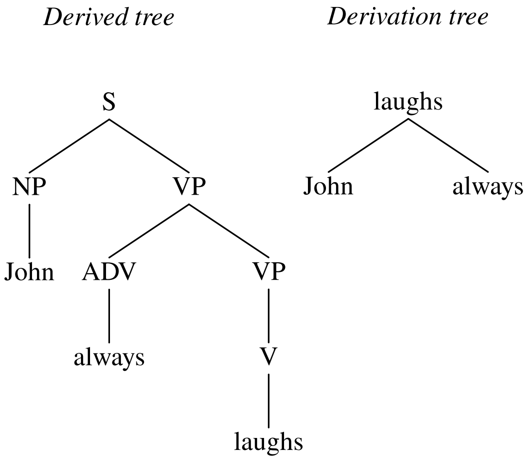


Figure 2: Derived tree and derivation tree for *John always laughs.*

In order to represent natural languages, TAGs are enriched with additional linguistic principles. First, a TAG for natural languages is *lexicalized* (Schabes, 1990), which means that each elementary tree has a lexical anchor (usually unique, but in some cases, there is more than one anchor). Second, the elementary trees of a lexicalized TAG (LTAG) represent extended projections of lexical items (the anchors) and encapsulate all syntactic arguments of the lexical anchor; that is, they contain slots (nonterminal leaves) for all arguments. Furthermore, elementary trees are minimal in the sense that only the arguments of the anchor are encapsulated; all recursion is factored away. This amounts to the *condition on elementary tree minimality* from (Frank, 1992). The tree for *laughs* in Figure 1, for example, contains only a nonterminal leaf for the subject NP (a substitution node), and there is no slot for a VP adjunct. The adverb *always* is added by adjunction at an internal node.

Because of these principles, in linguistic applications, combining two elementary trees by substitution or adjunction corresponds to the application of a predicate to an argument. The derivation tree then reflects the predicate-argument structure of the sentence. This is why most approaches to semantics in TAG use the derivation tree as an interface between syntax and semantics.

Feature structures are used by a variety of linguistic formalisms as a means for representing different levels of linguistic information. In a TAG, feature structures are associated with the nodes of elementary trees (K. V. Shanker, 1988) to provide an additional dimension to state linguistic generalizations.

Tree-adjoining languages fall into the class of mildly context-sensitive languages and as such are more powerful than context-free languages. The TAG formalism in general and lexicalized TAGs in particular, are well-suited for linguistics applications. It is shown that the properties of TAG allow the encapsulation of diverse syntactic phenomena in a very natural way. Furthermore, the possibility to convert a grammar in TAG formalism to other formalisms is open (*cf.* (Yoshinaga et al., 2003)). These caracteristics motivate us to choose TAG to model the Vietnamese grammar: on the one hand we try to adapt a generic parser to Vietnamese language, and on the other hand we try to create a reusable resource for the tasks concerning Vietnamese syntactic analysis and its evaluation.

In the next section we present our lexicalized tree-adjoining grammar for Vietnamese.

Word	Category	Meaning
trên	adjective	upper, above
	adverb, preposition	upper, on, over
	noun	the superior
trong	adjective	in, inside, internal
	preposition, conjunction	within
	noun	the interior

Table 1: Category mutations in Vietnamese

3. vnLTAG

An LTAG comprises a morphological and syntactic lexicon and a large repository of elementary trees. In order to take into account the reusability and possible multilingual applications, the lexicon of vnLTAG uses a tagset which is constructed from Vietnamese morpho-syntactic descriptors compatible with MULTEXT[2] (Multilingual Text Tools and Corpora), a series of projects whose goals are to develop standards and specifications for the encoding and processing of linguistic corpora for a wide variety of languages. Our lexicon implements the international standard ISO/DIS 24610-1[3] that provides a format to represent, store and exchange feature structures in natural language processing applications, for both the annotation and production of linguistic data. This standard also helps us build normalized morpho-syntactic annotations and describe the grammatical usage of Vietnamese lexical units. It is worth emphasizing that Vietnamese is an isolating language in which almost every simple word is monosyllabic and there is no morphological variation, and that all grammatical relations are determined by word order and tool words.

3.1. Categories and feature structures

The classification of grammatical categories for Vietnamese is still in debate amongst the linguistic community. The main difficulty comes from the ambiguity between grammatical roles for many words. The category mutation between nouns and verbs without any morphological variation is very frequent. In general, Vietnamese articles can be used as nouns, and the adjectives and prepositions can sometimes play the role of nouns. Table 1 gives some examples of such behaviours.

In the works of Nguyen (Nguyen et al., 2003; Nguyen et al., 2004b), a tagset for the morphosyntactic analysis of Vietnamese that is inspired from MULTEXT model was constructed. The definition of such a tagset is based on some principal criteria of the syntactic distribution. Some particular linguistic specificities of Vietnamese are also taken into account to build a two-level tagset. The first level tagset, that contains all major syntactic categories

[2] MULTEXT – http://www.lpl.univ-aix.fr/projects/multext/

[3] ISO subcommitee TC37/SC4 – http://www.tc37sc4.org/

No.	Category	Notation
0.	Noun	N
1.	Verb	V
2.	Adjective	A
3.	Pronoun	P
4.	Adverb	R
5.	Adposition	O
6.	Conjunction	C
7.	Determiner/Article	D
8.	Numeral	M
9.	Interjection	I
10.	Modal Particle	T
	Sentence	S
	Verbal Phrase	PredP
	Adjectival Phrase	PredP
	Prepositional Phrase	OP

Table 2: First level syntactic categories of Vietnamese

and phrases of Vietnamese, is given in Table 2. In this table, both the verbal phrase and the adjectival phrase are annotated PredP (Pred stands for Predicate), due to the fact that in Vietnamese, predicative sentences are expressed without any explicit copula verb.

We have defined several feature structures to represent and precise the linguistic information for the language. These feature structures are used in the syntactic dictionary and they are associated with elementary trees of the grammar. The grammar in general, and the set of feature structures in particular, is updated frequently, so the description of features structures in Table 3 may be not the latest implementation.

We have presented in (Nguyen et al., 2004a) a first work on the definition of an LTAG grammar for Vietnamese, that dealt with the case of noun phrases; that first part has since been improved, but due to lack of space we only present in the following section the additional results we have obtained for verb phrases.

3.2. Elementary trees for basic verb constructions

In our framework, we view all basic structures as being produced by a lexical item in the lexicon. In this framework, as in a TAG, the linguistic unit is the sentence. We have defined 21 basic verb phrase structures for Vietnamese to model the most frequently used Vietnamese simple sentences.

The first 13 structures are represented in the grammar by 12 initial trees corresponding to declarative sentences and complement clauses. In general, a verb is defined by its syntactic argument structure and the corresponding set of trees are associated with it. We refer to a given argument structure as a tree family (Abeillé, 2002). For example, Figure 3 shows the transitive structure and Figure 4 shows the complement clause structure of the language. Note that some feature structures are associated with the V nodes of the families to encode precise information about the verbs.

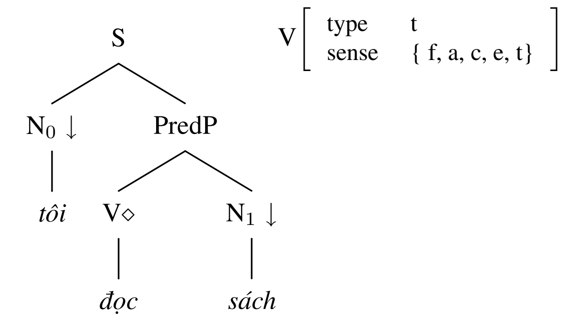

S
$N_0 \downarrow$
PredP
tôi
$V\diamond$
$N_1 \downarrow$
đọc
sách
$V\begin{bmatrix} \text{type} & \text{t} \\ \text{sense} & \{\text{ f, a, c, e, t}\} \end{bmatrix}$

Figure 3: Declarative transitive structure $\alpha n_0 V n_1$

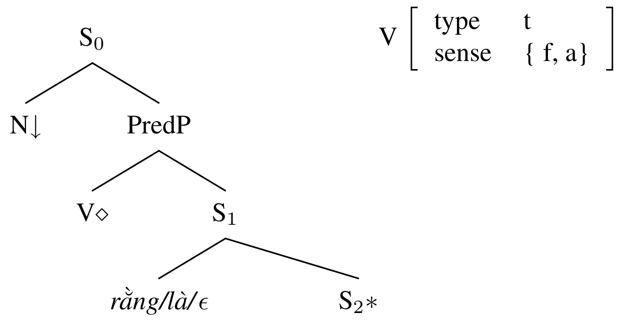

S_0
$N\downarrow$
PredP
$V\diamond$
S_1
rằng/là/ϵ
S_2*
$V\begin{bmatrix} \text{type} & \text{t} \\ \text{sense} & \{\text{ f, a}\} \end{bmatrix}$

Figure 4: Complement clause structure $\alpha n_0 V S$

The representation of a verb taking a sentential argument can be viewed as the composition of two sentential structures. The standard way of composing two structures in a TAG is to have one adjoined to the other. The adjunction node S_1* of the complement clause family permits one to generate complex sentences of any depth, for example:

- *Cô ấy đã đúng* [5];
- *Anh ấy tin [cô ấy đã đúng]* [6];
- *Nam cho rằng [anh ấy nói [cô ấy đã đúng]]* [7];

3.3. Optional complements construction

The current grammar contains 8 auxiliary families to represent optional structures or the modifiers. For example, auxiliary tree families shown in Figure 5 can allow pre and post modification of verbal phrases by adjoining onto them.

In Vietnamese, one can always add optional supplements to a verbal phrase to detail an action. Most often, these facultative complements give information about the time, the location and the manner of an action. Some instances for the optional suffix complement of a verbal phrase shown in Figure 6 are:

- time complement : *làm việc trong hai ngày liền*[8]
- location complement : *ngồi ở bãi cỏ* [9]
- manner complement : *in bằng kỹ thuật mới*[10]

A more detailed explanation about all the families and a large set of corresponding examples can be found in (Le, 2005).

[5] She was right

[6] He believes that she was right

[7] Nam thinks that he said that she was right

[8] work for two days

[9] sit on the grass

[10] print using a new technology

No.	Attribute	Values	Interpretation	Associated Nodes
0.	deg	+,–	degree	A, V, R
1.	human	+,–	human	N
2.	neg	+,–	negative	VP
3.	pers	1,2,3	person	N
4.	princ	+,–	principal	V
5.	copula	+,–	copula	V
6.	modif	+,–	with or without modifier	V, A, R
7.	type		type of categories	V, PredP, N
8.	sense	f,a,c,g,i,o,e,t,m[4]	sense of a lexeme	V, PredP

Table 3: List of feature structures of vnLTAG

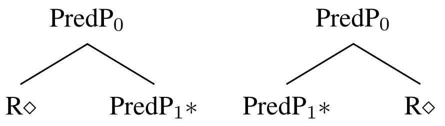


Figure 5: Auxiliary structures βRv and βvR

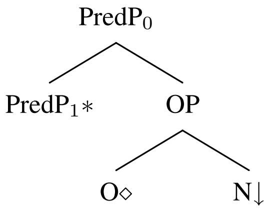


Figure 6: Suffix complement with an adposition structure βvOn

4. Implementation

In this section, we describe briefly the implementation choices for our grammar. We next present a parsing scheme for the grammar. Finally, we discuss some intial results of our promising approach for Vietnamese parsing.

4.1. Data format

The format we have chosen to represent the grammar is TAGML, an XML-based format that we first quickly present here before describing the format of our test suite.

4.1.1. TAGML format

We have adopted the TAGML format to represent the vnLTAG grammar. TAGML is a standard for the XML description of necessary resources used by LLP2, a LTAG parser that has been developed at LORIA for several years. TAGML is a effective format constructed on an XML Schema (XSD)[11] which is compatible with the international standard ISO/DIS 24610-1 for the representation, storage, and exchange of feature structures[12] Furthermore, the TAGML standard makes it possible to extend the access to elementary trees with the help of feature structures. Hereby is an example of the TAGML format used to define a feature structure and a tree family:

```
<!-- A feature structure definition -->
<fs id="_transitive" type="V">
 <f name="T">
  <string value="t"/>
 </f>
</fs>
```

```
<!-- A tree family definition -->
<tree id="VtNP">
 <fs>
  <f name="verbalFamily">
   <string value="VtNP"/>
  </f>
 </fs>
 <node cat="S" name="S">
  <node cat="N" name="N0" type="subst"/>
  <node cat="PredP" name="PredP">
   <node cat="V" name="V" type="anchor">
    <narg type="top">
     <fs feats="_transitive"/>
    </narg>
   </node>
   <node cat="N" name="N1" type="subst"/>
  </node>
 </node>
</tree>
```

4.1.2. Test suite format

The test suite data are stored in XML format. Each test item corresponds to a basic sentence construction that we considered during our work on Vietnamese grammar: each construction was simultaneously introduced in the grammar and in the test suite. The following example illustrates the XML format we have defined for that purpose.

[11] XML Schema – http://www.w3.org/XML/Schema

[12] http://www.tc37sc4.org/

```
<!-- A test item of the test suite -->
<ts id="VtNP" type="V">
 <ph main="s2" op="">
 <!-- phenomenon concerning s2 -->
  <struct cat="S">
   <struct cat="N">
    <lex id="tooi_1"/> <!-- I -->
    <!-- reference to the lexicon -->
   </struct>
  <struct cat="PredP">
   <struct cat="V" id="s2">
    <lex id="awn_1"/> <!-- eat -->
   </struct>
   <struct cat="N">
    <lex id="cowm_1"/> <!-- rice -->
   </struct>
  </struct>
 </ph>
 <ph main="s2" op="delete">
  <!-- ... -->
 </ph>
</ts>
```

4.2. Parser

We have adapted the syntactic parser LLP2 of LORIA to put the vnLTAG grammar into practice. LLP2 is a dedicated LTAG software that uses the TAGML format for the represenation of grammars. In addition to the main parsing module that implements an ascending algorithm, this parser is accompanied by several tools that allow not only to explore the morphologic lexicon and schema database but also to visualize intuitively all the possible lexeme–scheme associations of a parsing. These tools also give means to diagnose the cause of an analysis failure for a given phrase. Consequently, it makes up a cycle of grammar and tool development. Further information about TAGML and LLP2 is available at our website[13].

The parsing scheme for vnLTAG is shown on Figure 7.

A sentence is processed as follows:

- First, a Vietnamese tagger is used to tokenize the sentence into lexical units (or words) and associate with these words their possible morphosyntactic categories. The tagger makes use of a morphologic lexicon.

- The output of the tagger is then used as the input for the parser. In this phrase, a syntactic lexicon that contains elementary trees is used to help select tree families associated with tokenized words.

- Next, the parser analyses the data and the possible results of a parsing are given in the form of derivation trees and their corresponding derived trees.

- Finally, a tree visualizer may be used to show analysed trees.

Figure 8 gives a parsing result for the Vietnamese sentence "*Tôi tặng hoa cho người yêu*"[14]

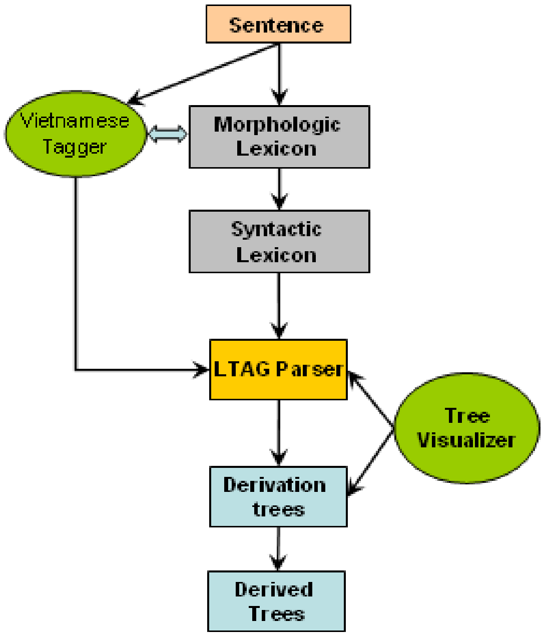


Figure 7: The parsing scheme for Vietnamese

4.3. Grammar evaluation

Evaluating a broad-coverage grammar is a difficult task, especially in the absence of a syntactically annotated reference corpus (or treebank) for Vietnamese. We could only perform a quick evaluation using the presented test suite.

Due to the method of construction of that test suite, which was carried on in parallel with the grammar definition, all phenomena are, naturally, taken into account by the grammar. The still limited vocabulary available to the parser did also not let many possibilities for ambiguities to appear, and all incorrect sentences were recognized as such. As biased as that first validation may seem, it is important to keep in mind the fact that grammatical rules and test cases were built from linguistic descriptions of the bases of Vietnamese, thus ensuring a core of functionality. The future developments of the system, and in particular the extension of the syntactic lexicon, will let us build more elaborate test cases, and perform more realistic evaluations.

The resources of vnLTAG (a small syntactic lexicon and elementary trees in TAGML format, as well as the test suite) and the parser are free for use and downloadable from the LORIA website[15].

5. Conclusion

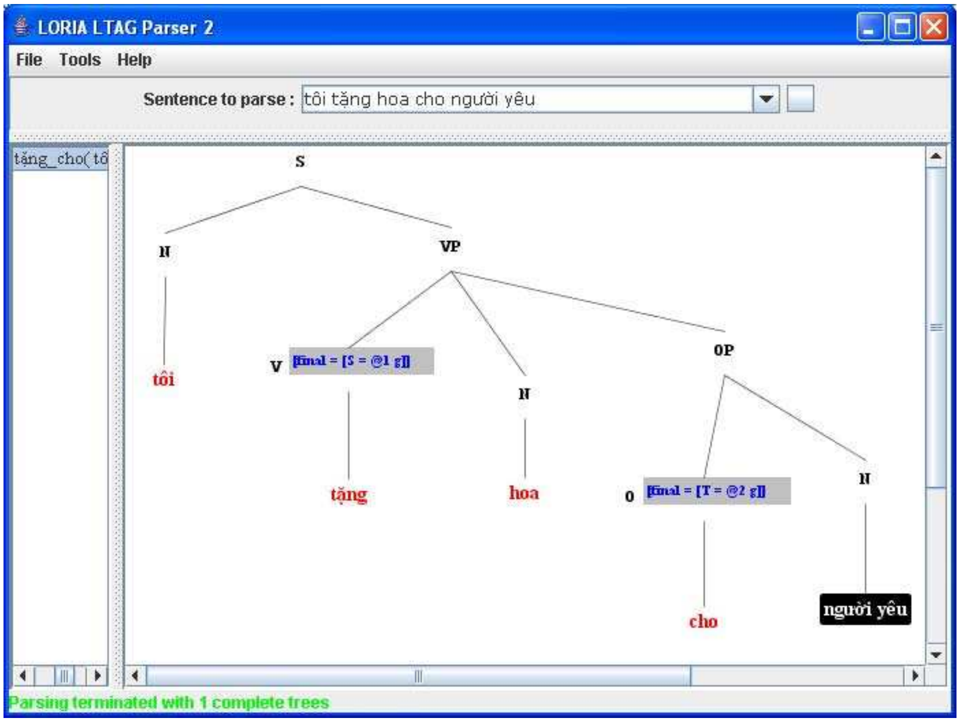
The choice of the LTAG formalism for parsing Vietnamese has both computational and linguistic advantages. The linguistic stipulations are minimized and the general organization of the grammar is simplified: all structures are stated in terms of surface structures, and there is a direct matching between the lexical information and the tree structures. The implementation of such a grammar leads to the adaptation of the LORIA LTAG parser for parsing Vietnamese.

Independently from the technical choice of using LTAG, our work also has the ambition of proposing a first formali-

[13] LLP2 – http://www.loria.fr/ azim/LLP2/help/fr/

[14] I give some flowers to my darling.

[15] vnLTAG parser – http://led.loria.fr/outils.php



Sentence to parse : tôi tặng hoa cho người yêu

Parsing terminated with 1 complete trees

Figure 8: Parsing result for the sentence “*Tôi tặng hoa cho người yêu*”

sation of Vietnamese grammar, and a first set of references for the evaluation of future works – notably in the shape of a comprehensive test suite.

The most immediately needed work is to complete the grammar by modelling adjective phrases and sentence-level modifiers (adverbs, modal particles, *etc.*). Once that is achieved, we can use the vnLTAG grammar as a tool to help for the construction of a Vietnamese Treebank, thus opening the way to the definition of actual broad-coverage grammars.

6. References

A. Abeillé. 2002. *Une grammaire électronique du français*. CNRS, Paris.

L. Balkan, K. Netter, D. Arnold, and S. Meijer. 1994. TSNLP — Test Suites for Natural Language Processing. In *Proceedings of Language Engineering Convention*, pages 17–22, Edinburgh. ELSNET.

R. Frank. 1992. *Syntactic Locality and Tree Adjoining Grammar: Grammatical Acquision and Processing Perspectives*. Ph.D. thesis, University of Pennsylvania.

A. Joshi, L. S. Levy, and M. Takahashi. 1975. Tree adjunct grammars. *Journal of the Computer and System Sciences*.

A. Joshi, 1985. *Tree Adjoining Grammars: How much context sensitive is required to provide a resonable structural description*, pages 206–250. Cambridge University Press, Cambridge, England.

A. Joshi K. V. Shanker. 1988. Feature-structure based tree adjoining grammar. In *Proceedings of COLING 12*, pages 714–719, Budapest.

A. Kroch and A. Joshi. 1985. The linguistic relevance of tree adjoining grammars. Technical report, University of Pennsylvania.

H. P. Le. 2005. Vers une grammaire électronique du vietnamien. Master’s thesis, Institut de la Francophonie pour l’Informatique, Hanoi, Vietnam.

T. M. H. Nguyen, L. Romary, and X. L. Vu. 2003. Une étude de cas pour l’étiquettage morpho-syntaxique de textes vietnamiens. In *TALN 2003*, Batz-sur-Mer, France.

T. B. Nguyen, T. M. H. Nguyen, L. Romary, and X. L. Vu. 2004a. Developing tools and building linguistic resources for vietnamese morpho-syntactic processing. In *4th International Language Resources and Evaluation Conference (LREC’04)*, Lisbon, Portugal.

T. B. Nguyen, T. M. H. Nguyen, L. Romary, and X. L. Vu. 2004b. Lexical descriptions for vietnamese language processing. In *ALR–04, Workshop on Asian Language Resources*, Hainan, China.

Y. Schabes. 1990. *Mathematical and Computational Aspects of Lexicalized Grammars*. Ph.D. thesis, University of Pennsylvania.

N. Yoshinaga, Y. Miyao, K. Torisawa, and J. Tsujii. 2003. Parsing comparison across grammar formalisms using strongly equivalent grammars. Comparison of LTAG and HPSG parsers: A case study". *Traitement Automatique des Langues – Evolution en analyse syntaxique*, 44:15–39.