HAL open science

Building a Large Syntactically-Annotated Corpus of Vietnamese

Hong Phuong Le, Thi Minh Huyen Nguyen, Phuong Thai Nguyen, Xuan Luong Vu, van Hiep Nguyen

To cite this version:

Hong Phuong Le, Thi Minh Huyen Nguyen, Phuong Thai Nguyen, Xuan Luong Vu, van Hiep Nguyen. Building a Large Syntactically-Annotated Corpus of Vietnamese. The Third Linguistic Annotation Workshop (The LAW III), Aug 2009, Singapour, Singapore. 6p. inria-00421103v1

HAL Id: inria-00421103

https://inria.hal.science/inria-00421103v1

Submitted on 12 Nov 2009 (v1), last revised 15 Dec 2009 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Building a Large Syntactically-Annotated Corpus of Vietnamese

Abstract

Treebank is an important resource for both research and application of natural language processing. For Vietnamese, we still lack of such kind of corpora. This paper presents up-to-date results of a project for Vietnamese treebank construction. Since Vietnamese is an isolated language and has no word delimiter, there are many ambiguities in sentence analysis. We systematically applied a lot of linguistic techniques to handle such ambiguities. Annotators are supported by automatic-labeling tools and a tree-editor tool. Raw texts are extracted from Youth, an online Vietnamese daily newspaper. The current annotation agreement is around 90 percent.

1 Introduction

Treebanks are used for training syntactic parsers, part-of-speech taggers, and word segmenters. These systems then can be used for applications such as information extraction, machine translation, question answering, and text summarization. Treebanks are also useful for linguistic studies, for example the extraction of syntactic patterns or the investigation of linguistic phenomena. Recently, treebanks and other large corpora become more important since the development of powerful machine learning methods.

Vietnamese is an isolated language. There is no word delimiter in Vietnamese. The smallest unit in the construction of words is syllables. Words can be single or compound. Vietnamese script is invented based on Latin alphabet which the expansion includes accent characters and stressed accents.

Since Vietnamese word order is quite fixed, we choose to use constituency representation of syntactic structures. For languages with freer word order such as Japanese or Szech, dependency representation is more suitable. We apply annotation scheme proposed by Marcus et al. (1993). This approach has been successfully applied to a number of languages such as English, Chinese, Arabic, etc.

For Vietnamese, there are three annotation levels including word segmentation, POS tagging, and syntactic labeling. Word segmentation identifies word boundary in sentences. POS tagging assigns correct POS tags to words. Syntactic labeling recognizes both phrase-structure tags and functional tags. Our main target is to build a corpus of 10,000 syntactically-annotated sentences (trees) and an additional POS tagged data set of 10,000 sentences. Treebank construction is a very complicated task including major phases: investigation, guideline preparation, building tools, raw text collection, and annotation. This is a repeated process especially three phases: annotation, guideline revision, and tool upgrade. Raw texts are collected from a newspaper source, the Youth online daily newspaper, with a number of topics including social and politics. We completed about 9,500 trees and 10,000 POS tagged sentences.

In order to deal with ambiguities occurring at various levels of annotation, we systematically applied linguistic analysis techniques such as deletion, insertion, substitution, questioning, transformation, etc. Notions for analysis techniques are described in guideline. These techniques are originated in literatures or proposed by our group. They are described with examples, arguments, and alternatives. For automatic labeling tools, we used advanced machine learning methods such as CRFs for POS tagging or LPCFGs for syntactic parsing. These tools helped us speed up labeling process. Besides, tree editor was also very helpful.

Our treebank project is a branch project of a national project which aims to develop basic resources and tools for Vietnamese language and speech processing. This national project is called VLSP. In addition to treebank, other text-processing resources and tools include: Vietnamese machine readable dictionary, English-

Vietnamese parallel corpus, word segmenter, POS tagger, chunker, and parser. Treebank and tools are closely related. Tools are trained using treebank data, and then they can be used in treebank construction.

The rest of this paper is organized as follow: First, we represent an overview of important aspects in treebank construction. Second, we present issues in Vietnamese word segmentation problem. Third, POS tagging and syntactic parsing are described. Fourth, tools and annotation process are represented. Fifth, we present annotation agreement evaluation. And last, some conclusion is drawn.

2 Existing Treebanks

2.1 Treebank Construction Process

There are a number of major steps:

- Investigation: Choosing a suitable annotation approach for the considered language. Determine what to do and obstacles. This step concerns linguistics, treebank, and computational linguistics.
- Guideline preparation: Crucial to the treebank quality. This document describes tag sets and annotation guidelines for syntactic structures. Concrete examples are given.
- Tool preparation: both automatic labeling tools and a tree editor
- Raw text collection: from newspaper, book, or other sources
- Data annotation: this is the most time-consuming task since data often requires a number of revisions.

2.2 Annotation Schema

Annotation schemas can be divided into two types in which the first one aims to represent phrase structure of sentences while the second one focuses on dependencies between words. Rambow and Joshi (1997) shown that constituency tree can be converted into dependency tree but not the inverse direction. Besides, divergences between annotation schemas are also caused by differences in linguistic properties. For languages without word delimiter, word segmentation problem is a non-trivial problem. Inflectional languages require that morphological information is encoded in POS tags. Languages with free word order require that long distance dependencies are modeled in tree structures.

Constituency-based annotation schemas often share the following properties:

- Annotation levels: word segmentation, POS tagging, syntactic labeling. Where word segmentation is important for languages without word delimiter like Vietnamese.
- Syntactic labels: phrase structure tags and functional tags (including semantic role tags)
- Deep structures are represented by using null elements: useful for several phenomena such as wh-movement, topic, passive voice, etc.
- Basic syntactic relations are identified: head-modifier, coordination, predicate, complement, adjunct, topic, etc.

2.3 Guideline Preparation

A good annotation guideline should satisfy a number of requirements. First, the specificity of tags should be adequate, not too specific nor too general. Second, guideline should be consistent in both theoretical and practical aspects. Third, guideline should be neutral with linguistic theories. And last, a design should be general and easy to extend by adding new tags, splitting existing tags, etc.

When the early version of a guideline is being developed, a design group analyzes sentences collected from grammar books. This guideline version covers only basic grammatical phenomena. Then the guideline is incrementally extended to cover real text. Linguistic phenomena in practice are far more complex than basic patterns in books. Therefore the guideline will be modified and upgraded through the development of treebank projects.

3 Word Segmentation

There are many approaches to word definition, for example based on morphology, based on syntax, based on semantics, or linguistic comparison. We consider words as syntactic atoms (Sciullo and Williams, 1987) according to the sense that it is impossible to analyze word structure using syntactic rules, or that words are the smallest unit which is syntactically independent. We choose this criterion partly because the first application of word segmentation is for syntactic analysis (build trees).

According to application view, machine translation researchers may argue that Vietnamese words and foreign words should match each other. The problem is that there are so many possible foreign languages which are different in vo-

cabulary. Dictionary editors may want to extract phrases from text which need to be explained in meaning. For this application, syntactic parsers can be used as tool for editors. Parsers can extract candidates for phrase/word entry.

The following word types are considered in word segmentation phase:

- Single words
- Compound words
- Repeated words
- Idioms
- Proper names
- Date/time, number expressions
- Foreign words
- Abbreviations

Word segmentation ambiguity is the major problem annotators have to be deal with. Suppose that three words "nhà cửa", "sắc đẹp", and "hiệu sách" are being considered. Annotators need to identify these combinations as words in:

a. Nhà cửa bề bộn quá
b. Cô ấy giữ gìn sắc đẹp.
c. Ngoài hiệu sách có bán cuốn này

And not words in:

a. Ở nhà cửa ngõ chẳng đóng gì cả.
b. Bức này màu sắc đẹp hơn.
c. Ngoài cửa hiệu sách báo bày la liệt.

We used dictionaries as a reference. In practice, we consider dictionary words as candidate for word segmentation and make decision using context.

4 POS Tagging and Syntactic Annotation Guidelines

4.1 POS Tag Set

For European languages, word classes closely related to morphological aspects such as gender, number, case, etc. For Vietnamese, words are often classified based on their combination ability, their syntactic functions, and their general meaning. We choose first two criteria, combination ability and syntactic function, for POS tag set design. Therefore our POS tag set will not contain morphological information (number, tense, etc.), sub-categorization information (transitive/intransitive verbs, verbs followed by clauses, etc.), and semantic information.

4.2 Syntactic Tag Set

Our tag set contains three tag types: constituency tags, functional tags, and null-element tags. We use the tag H to label phrase head. If a phrase has more than one head, connected by coordination conjunctions or commas, then all heads are labeled with H tag. Other treebanks often does not use head tag. Therefore researchers on syntactic parsing (Collins, 1999) used heuristic rules to determine CFG rules' head. Machine learning methods also can be used (Chiang and Bikel, 2002). Null elements are often used for adjective clauses, ellipsis, passive voice, and topic.

4.3 Sentence and Phrase Analysis Techniques

Annotation of real text requires various techniques to be applied. Ambiguity may occur in many steps of analysis such as determining phrase's head, discriminating between possible complements, discriminating between adjuncts and other sentence elements, etc. Sentence analysis techniques include deletion, substitution, insertion, transformation, questioning. These techniques exploit contextual information, word combination, word order, and functional words to disambiguation between possible structures. Table 1 shows several examples of sentence analysis techniques.

No	Problems, techniques, and examples
1	Head word determination
	Deletion
	một đàn gà con mới nở... →một đàn gà con →một đàn gà →đàn gà con đàn gà
2	Verb sub-categorization discrimination between S and NP-VP
	Passive transformation
	Tôi bầu ông ấy làm hiệu trưởng. →Ông ấy được tôi bầu làm hiệu trưởng. (NP-VP) Anh ấy kể công ty đang làm ăn phát đạt. ~~→Công ty được anh ấy kể đang làm ăn phát đạt.~~ (S)
3	Discrimination between "là" NP and "là" S
	Insertion of "mà"
	Tôi là người đặt cược ít nhất → Tôi là người mà đặt cược ít nhất
4	Discrimination between "vì" NP and "vì" S
	Insertion of "ấy", "đó", "này", etc.
	Vì chuyện quá lạ nên anh Bình vẫn chưa tin. ("chuyện quá lạ" should be labeled S)
5	Discrimination between complement and predicate
	Insertion of adverbs such as "đã", "đang", "sẽ", etc.

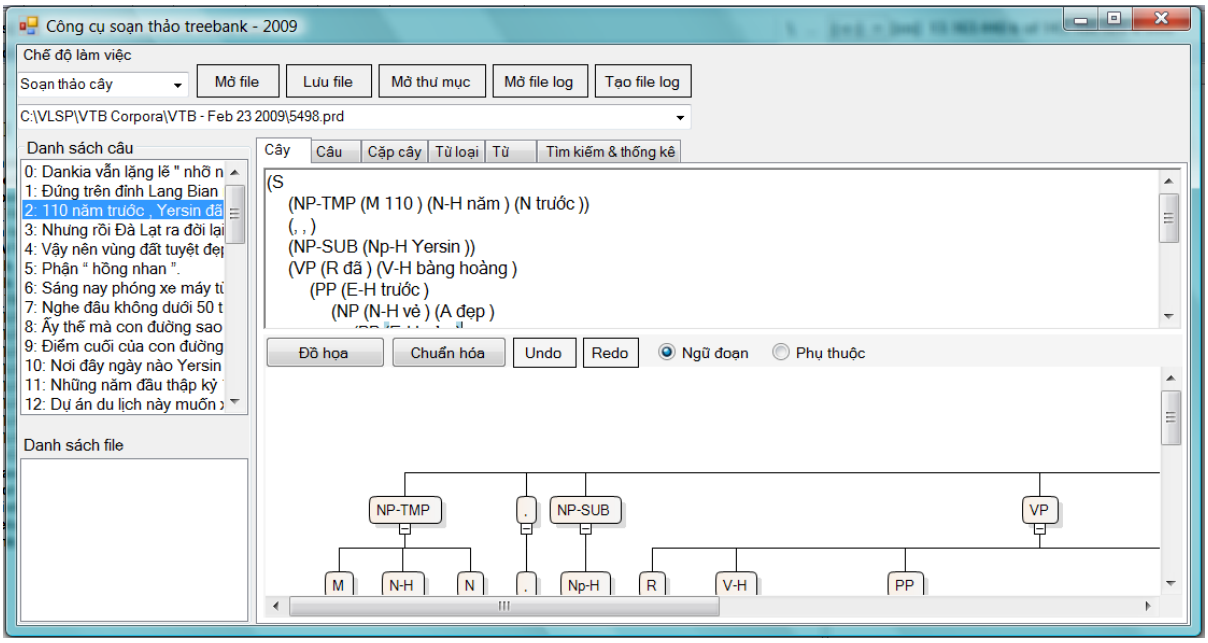

Figure 1. Treebank editor

	Cô gái tên Phạm Thị Thanh Th. , 19 tuổi , quê ở Cần Thơ học chưa hết lớp 7. →Cô gái tên Phạm Thị Thanh Th. , 19 tuổi , quê ở Cần Thơ mới chỉ học chưa hết lớp 7. The phrase "tên Phạm Thị … Cần Thơ" is an adjunct of "cô gái" since we can not insert tense and aspect words (such as "đã", "đang", etc.) between that phrase and "cô gái"
6	Labeling PRP
	Insertion of "để"
	Nó đi làm kiếm tiền nuôi em. →Nó đi làm để kiếm tiền nuôi em.
7	Discrimination of adjuncts and other sentence elements
	Reordering
	Ở quê, mẹ đang cấy lúa. →Mẹ ở quê đang cấy lúa. →Mẹ đang cấy lúa ở quê.

Table 1. Examples of disambiguation techniques

4.4 Linguistics Issues

The problem of treebank construction can be considered as an application of linguistic theories though treebanks can also be used for linguistic studies. However, there are still disagreements among linguists as to solutions for many linguistic issues. For example, that the classifier noun is noun phrase's head or pre-modifier is controversial. Another example, Vietnamese sentence structure is subject-predicate or topic-comment is also controversial. Our treebank more relies on subject-predicate structure. Moreover, we choose linguistic solutions most appropriate to our design.

5 Tools

We designed a tool for supporting annotators in most all phases of the annotation process. Main functions of our editor are as follows:

- Edit and view trees in both text mode and graphical mode
- View log files, highlight modifications
- Search by words or syntactic patterns
- Predict errors (edit, spell, or syntax)
- Compute annotation agreement and highlight differences
- Compute several kinds of statistics

For encoding the treebank, we have developed an exchange format named vnSynAF, a syntactic annotation framework which is conformed to the standard framework SynAF of ISO. The framework SynAF is built on top of an XML-based annotation scheme which is recommended by ISO for the encoding of treebanks[1]. Our tool also supports bracketing representation (or Lisp style) of Penn English Treebank. These formats can be converted into each other.

For the task of word segmentation, we used vnTokenizer, a highly accurate segmenter which uses a hybrid approach to automatically tokenize

[1] ISO/CD/24615, Language Resource Management-Syntactic Annotation Framework (SynAF) TC37/SC 4 N421, 22th Aug 2007, http://tc37sc4.org/documents

Vietnamese text. The approach combines both finite-state automata technique, regular expression parsing, and the maximal-matching strategy which is augmented by statistical methods to resolve ambiguities of segmentation (Phuong et al., 2008).

We used JVnTagger, a POS tagger based on Conditional Random Fields (Lafferty et al., 2001) and Maximum Entropy (Berger et al., 1996). This tagger is also developed under supported of VLSP project. Training data size is 10,000 sentences. Experiments with 5-fold cross validation showed that F1 scores for CRFs and Maxent are 90.40% and 91.03% respectively.

A syntactic parser based on Lexicalized Probabilistic Context-free Grammars (LPCFGs) is another tool we used. Another group in VLSP customized Bikel's parser[2] for parsing Vietnamese text. This parser is a well designed and easy to adapt to new languages. The group implemented a Vietnamese language package which handles treebank, training, finding head of CFG rules, and word features. This parser can output text with constituent tags only or both constituent tags and functional tags.

6 Annotation Process and Agreement

There are three annotation levels: word segmentation, POS tagging, and syntactic labeling. Since the word segmentation tool had been available before the start of our project, it was used for the first annotation level (word segmentation) immediately. As to the other annotation levels (POS tagging and syntactic parsing), first several thousand sentences were labeled manually. After that a POS tagger and a parser are trained bimonthly, then the annotation task becomes semi-automatic. According to our annotation process, each sentence is annotated and revised by at least two annotators. The first annotator labels raw sentences or revises automatically-analyzed sentences. Then the second annotator revises the output of the first annotator. In addition, we also check corpus by syntactic phenomena, for example direction words, questions, etc. This process is supported by tool. So there are many sentences which are revised more than twice.

Table 2 shows a number of important corpus statistics such as sentence count, word count, and syllable count for two data sets. We completed the POS tagged data set and will complete the syntactically-labeled data set soon. The average sentence length is about 21.6 words.

Data set	Sentences	Words	Syllables
POS tagged	10,368	210,393	255,237
Syntactically labeled	9,633	208,406	251,696

Table 2. Corpus statistics

Annotation agreement measures how similar two texts annotated independently by different annotators are. Since this problem is similar to parsing evaluation, we use parseval measure. First, syntactic constituents in the form (i, j, label) are extracted from syntactic trees. Then tree comparison problem is transformed into constituent comparison. We can compute three kinds of measurement: constituent and function similarity, constituent similarity, and bracket similarity. By using this method, we can evaluate both overall agreement and constituency agreement.

Annotation agreement A between two annotators can be computed as follows:

$$A = \frac{2 \times C}{C_1 + C_2}$$

where C_1 is the number of constituents in the first annotator's data set, C_2 is the number of constituents in the second annotator's data set, and C is the number of identical constituents. Table 3 shows and example of constituent extraction from trees. From Table 3, we can compute: C_1=6; C_2=7; C=6; A=12/13=0.92 .

1st annotator	2nd annotator
(S (NP (Np Hằng)) (VP (V ngắm) (NP (N mưa)) (PP (E trong) (NP (N công viên)))) (. .))	(S (NP (Np Hằng)) (VP (V ngắm) (NP (NP (N mưa)) (PP (E trong) (NP (N công viên))))) (. .))
(1,6,S); (1,1,NP); (2,5,VP); (3,3,NP); (4,5, PP); (5,5,NP)	(1,6,S); (1,1,NP); (2,5,VP); (3,3,NP); *(3,5,NP)*; (4,5, PP); (5,5,NP)

Table 3. Constituent extraction from trees

We carried out an experiment involving 3 annotators. They annotated 100 sentences and the result is shown in Table 4.

Test	A1-A2	A2-A3	A3-A1
Full tags	90.32%	91.26%	90.71%
Constituent tags	92.40%	93.57%	91.92%
No tags	95.24%	96.33%	95.48%

Table 4. Annotation agreement

[2] http://www.cis.upenn.edu/~dbikel/software.html

7 Conclusions

In this paper, we presented our most up-to-date results on Vietnamese treebank construction. This project is coming to final stage. We continue to annotate more text, revise data by syntactic phenomenon and feedback from users. We also use statistical techniques to analyze treebank data to find out errors and fix them. We intend to publish this data on LDC this year.

Reference

Diệp Quang Ban. 2005. Ngữ pháp tiếng Việt (2 tập). *NXB Giáo dục*.

Vũ Tiến Dũng. Tiếng Việt và ngôn ngữ học hiện đại sơ khảo về cú pháp. 2003. VIET Stuttgart – Germany.

Cao Xuân Hạo. 2006. Tiếng Việt sơ thảo ngữ pháp chức năng. *NXB Khoa học Xã hội*.

Nguyễn Văn Hiệp. Vài nét về lịch sử nghiên cứu cú pháp tiếng Việt. Tạp chí Ngôn ngữ, Hà Nội, số 10/2002.

Nguyễn Kim Thản. 2008. Cơ sở ngữ pháp tiếng Việt. *NXB Khoa học Xã hội*.

Nguyễn Minh Thuyết và Nguyễn Văn Hiệp. 1999. Thành phần câu tiếng Việt. *NXB ĐHQG Hà Nội*.

Ủy ban Khoa học Xã hội Việt Nam. 1983. Ngữ pháp tiếng Việt. NXB Khoa học Xã hội.

Adam Berger, Stephen D. Pietra, and Vincent D. Pietra. 1996. A maximum entropy approach to natural language processing. Computational Linguistics, (22-1).

Sabine Brants et al. The TIGER Treebank. 2003. COLING.

David Chiang and Daniel M. Bikel. 2002. Recovering Latent Information in Treebanks. COLING.

Michael Collins. 1999. Head-Driven Statistical Models for Natural Language Parsing. PhD thesis, University of Pennsylvania.

Chung-hye Han et al. Development and Evaluation of a Korean Treebank and its Application to NLP. 2002. LREC.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. ICML.

Mitchell P. Marcus et al. Building a Large Annotated Corpus of English: The Penn Treebank. 1993. Computational Linguistics.

L. H. Phuong, N. T. M. Huyen, R. Azim, H. T. Vinh. A hybrid approach to word segmentation of Vietnamese texts. Proceedings of the 2nd International Conference on Language and Automata Theory and Applications, Springer LNCS 5196, Tarragona, Spain, 2008.

Owen Rambow and Aravind K. Joshi. 1997. A formal look at dependency grammars and phrase structure grammars with special consideration of word-order phenomena. In L. Wenner, editor, Recent Trends in Meaning-Text Theory. John Benjamin, Amsterdam, Philadelphia.

Anna M.D. Sciullo and Edwin Williams. 1987. On the definition of word. The MIT Press.

Peter Sells. Lectures on Contemporary Syntactic Theories. 1987. CSLI.

Fei Xia et al. Developing Guidelines and Ensuring Consistency for Chinese Text Annotation. 2000. COLING.

Nianwen Xue et al. Building a Large-Scale Annotated Chinese Corpus. 2002. COLING.