Typed Dependency Relations for Syntactic Analysis of Thai Sentences*

Siripong Potisuk

Department of Electrical and Computer Engineering
The Citadel, the Military College of South Carolina
171 Moultrie Street
Charleston, South Carolina 29409 USA
siripong.potisuk@citadel.edu

Abstract. This paper describes a preliminary effort in identifying many different types of relations among words in Thai sentences based on dependency grammar. The relation is represented as a triple containing the pair of words and their relation. So far, the current representation contains 35 grammatical relations. The dependencies are all binary relations. That is, a grammatical relation holds between a governor and a dependent. The analysis makes use of the Thai "Orchid" corpus part-of-speech tags and the Stanford typed dependencies definitions.

Keywords: Dependency grammar, Thai syntactic analysis.

1 Introduction

Language modeling is one of the many important aspects in natural (both written and spoken) language processing by computer. For example, in a spoken language understanding system, a good language model not only improves the accuracy of low-level acoustic models of speech, but also reduces task perplexity (the average number of choices at any decision point) by making better use of high-level knowledge sources including prosodic, syntactic, semantic, and pragmatic knowledge sources. A language model often consists of a grammar written using some formalism which is applied to a sentence by utilizing some sort of parsing algorithm.

One popular example is a set of context-free grammar (CFG) production rules, which is based on a phrase-structure representation of syntax, can be used to parse sentences in the language defined by that grammar. Another example is constraint dependency grammar (CDG) proposed by Maruyama. CDG parsers rule out ungrammatical sentences by propagating constraints. Constraints are developed based on a dependency-based representation of syntax. Some parsing algorithms combine phrase-structure and dependency grammars. In the next subsection, we present a contrastive description of the two major approaches to representing syntax of natural languages in order to motivate our choice of dependency grammar for parsing Thai sentences.

1.1 Dependency vs. Phrase-structure Grammar

In the theory of the syntax of natural languages, there are currently two major methods of representing the syntactic structure of natural languages: dependency grammar and phrase-structure (constituency) grammar. There has been no third approach developed although combinations of the two major methods above have been used, e.g., lexical-functional grammar, case grammar, relational grammar, word grammar, etc..

* The author would like to thank the Citadel Foundation for its support in the form of a presentation grant.

As a formal syntactic representation, dependencies have been studied and explored for centuries by traditional syntacticians of European, Classical, and Slavic languages. Lucien Tesnière was credited as the first syntacticians who formalized and laid the groundwork for subsequent investigations of the theory. Unfortunately, the dependency formalism has not gained great popularity among syntacticians who favor constituency grammar. Constituency or phrase-structure grammar was formulated in North America in the early 1930's by Leonard Bloomfield, primarily for describing the syntax of English. The theory was seriously advanced by Noam Chomsky, and his transformational-generative approach has been accepted throughout the world. Phrase-structure syntax gradually forced dependency syntax into relative obscurity. Nevertheless, there have been some attempts to defend the use of dependency syntax, and several linguist have contributed to this cause. For example Mel' čuk presented an argument for the case of dependency formalism and bravely claimed that "dependencies are much better suited to the description of syntactic structure (of whatever nature) than constituency is." His contrastive description of the two methods is summarized next.

A dependency grammar describes the syntactic structure of a sentence by using a dependency tree (D-tree) to establish dependencies among words in terms of head and dependents. A D-tree shows a relational characteristic of the syntactic representation in the form of hierarchical links between items, i.e., which items are related to which other items and in which way. On the other hand, a phrase-structure grammar uses a phrase-structure tree (PS-tree) to describe the groupings of words into the so-called *constituents* at different levels of sentence construction. A PS-tree shows which items go together with other items to form tight units of a higher-order, a distributional characteristic of a grouping within a larger grouping.

A *tree* is a network consisting of nodes which are linked in a tree-like structure (i.e., with a stem and branches). In a syntactic tree, a node which represents a word or lexical item, the smallest syntactic unit, is called a terminal node; a node which represents an abstract syntactic grouping or phrase, such as noun phrase (NP), verb phrase (VP), prepositional phrase (PP), etc., is called a non-terminal node.

A D-tree contains only terminal nodes; no abstract representation of syntactic groupings is used. On the contrary, a PS-tree contains both terminal and non-terminal nodes; most nodes are, however, non-terminals. This hierarchical representation in terms of terminals and non-terminals in a PS-tree leads to the notion of syntactic class membership of an item (i.e., the categorization of belonging to an NP, VP, etc.). Syntactic class membership is a way of labeling syntactic roles in a PS-tree because a PS-tree does not and cannot specify the types of syntactic links existing between two items in a natural and explicit way. In contrast to PS-tree, class membership is not specified in a D-tree. Instead, a D-tree puts a particular emphasis on specifying in detail the type of any syntactic relation between two related items. Such syntactic relations are, for example, predicative, determinative, coordinative relations, etc. In addition, in terms of the ordering of nodes, nodes must be ordered linearly in a PS-tree. In a D-tree, however, nodes are not necessarily in a linear order.

From the above contrastive description of the two approaches to representing the syntax of natural languages, one can draw the following conclusion. The phrase-structure representation is suitable for languages like English, which have a rigid word order and a near absence of syntactically driven morphology. On the other hand, the dependency representation is suitable for languages like Latin or Russian, which feature an incredibly flexible (but far from arbitrary) word order and very rich systems of morphological markings. Word arrangements and inflectional affixes are obviously contingent upon relations between words rather than upon constituencies. In this paper, we argue for the choice of dependency representation of grammar for Thai.

1.2 Thai Language Characteristics and Related Researches

Thai is the official language of Thailand, a country in the Southeast Asian region. The language is spoken by approximately 65 million people throughout different parts of the country. The written form is used in school and in carrying out official business.

Difficulties in parsing Thai sentences arise for the following reasons. First, Thai sentences do not contain delimiters or blanks between words. Unlike English, Thai words in a sentence are not flanked by a blank space. Words are concatenated to form a phrase or sentence without explicit word delimiters. This creates a problem for the syntactic analysis of Thai sentences because most parsers operate on words as the smallest syntactic unit in a sentence. To overcome this problem, a word segmentation module must be added to the front end of the parser. This solution, in turn, creates a new problem. Instead of analyzing a single sentence, a parser must now analyze multiple sentence hypotheses comprising a combination of all possible words generated by the word segmentation algorithm.

Secondly, Thai words lack inflectional and derivational affixes. Since words in Thai do not inflect to indicate their syntactic function, the position of a word in a sentence alone shows its syntactic function. Hence, syntactic relationships are primarily determined by word order, and structural ambiguity often arises.

Thirdly, inconsistent ordering relations within and across phrasal categories characterize Thai sentences. While a noun, the head of a noun phrase, always precedes its modifying adjectives and determiners, the verb phrase exhibits less consistency. Although a verb, the head of the verb phrase, always precedes its object, its modifying auxiliaries can either precede or follow it. In addition, constituents which optionally occur with the head in both noun and verb phrases, such as determiners and quantifiers, tend to be less consistent in their ordering.

Lastly, Thai sentences sometimes contain discontinuous sentence constituents in their construction. In grammatical analysis, discontinuity refers to the splitting of a construction by insertion of another grammatical unit. In other words, discontinuity occurs when the elements which make up the constituents are interrupted by elements of another constituent in a sentence.

Research on the syntactic analysis of Thai sentences by computer has been carried out over a decade. Thai grammars have been developed utilizing various grammar formalisms based on the two major theories of syntax, namely dependency and phrase-structure grammar, or their combination. However, the most popular formalism has been the phrase-structure representation of syntax.

In recent years, dependency-based models have come to play an increasingly important role in the field of natural language parsing. First of all, dependency relations have proven very useful for statistical disambiguation in more traditional constituency-based parsing. In addition, there seems to be a growing interest in parsing systems that produce dependency structures rather than traditional phrase structure representations. Thus, it is our goal to try to apply dependency grammar to the wide-coverage syntactic analysis of Thai sentences. And, as a result, an efficient dependency-based parsing system can be developed for Thai. Also, the motivation for our choice of dependency grammar to overcome the difficulties in parsing Thai, instead of phrase-structure grammar mentioned in section 1, stems from the fact that it appears that Thai syntax might be better described by the former representation.

To the best of our knowledge, Aroonmanakul (1990) was the first to develop a deterministic parser called CUPARSE based on a dependency representation of syntax. The parser uses a chart as its central data structure. The implementation was based on 31 dependency relations. Potisuk (1996) proposed an alternative formalism called Constraint Dependency Grammar (CDG). CDG parsers rule out ungrammatical sentences by propagating constraints developed based on a dependency-based representation of syntax. It is claimed that CDG is capable of efficiently analyzing free-order languages because order between constituents is not a requirement of the grammatical formalism. Since Thai exhibits significant word order variation, using phrase-structure models to describe Thai is cumbersome because numerous rules would

be needed to cover all possible configurations of a constituent. However, the analysis of possible dependencies relations was not specified. The most recent manual for annotating dependency relations for Thai was proposed by Sudprasert (2008). He posited a total of 30 relations comprising two groups of relations: 12 types of complements and 18 types of modifiers. The next section describes our proposed dependency representation for Thai sentences.

2 Proposed Dependency Representation Manual

The relation is represented as a triple containing the pair of words and their relation. So far, the current representation contains 35 grammatical relations. The dependencies are all binary relations. That is, a grammatical relation holds between a governor and a dependent. The analysis makes use of the Thai "Orchid" corpus part-of-speech tags (1999). Also, the definition of each relation follows that of the Stanford typed dependencies manual (de Marneffe and Manning, 2008). The following are listings of the 35 grammatical relations in alphabetical order according to the dependency abbreviate names.

abbrev: abbreviation modifier
An abbreviation modifier of an NP is a parenthesized NP that serves to abbreviate the NP (or to define an abbreviation).

กระทรวงกลาโหม (กห.)	abbrev**(กระทรวง, กห.)**
Defense Ministry (kor hor)	abbrev(Ministry, kor hor)

advcl: adverbial clause modifier
An adverbial clause modifier of a VP is a clause modifying the verb (temporal clause, consequence, conditional clause, etc.).

ถ้าฝนตกเขาจะไม่ออกไปข้างนอก	advcl**(ออก, ตก)**
If it rains, he will not go outside.	advcl(go, rains)

advmod: adverbial modifier
An adverbial modifier of a word is a (non-clausal) adverb or ADVP that serves to modify the meaning of the word.

ฉันชอบสุนัขมาก	advmod**(ชอบ, มาก)**
I like dogs a lot.	advmod(like, a lot)

agent: agent
An agent is the complement of a passive verb. It either follows the auxiliary or a group of particles indicating passive voice.

เขาถูกพวกผู้ก่อการร้ายจับตัวไป	agent**(จับตัว, ผู้ก่อการร้าย)**
He was kidnapped by terrorists	agent(kidnapped, terrorists)

amod: adjectival modifier
An adjectival modifier of an NP is any adjectival phrase that serves to modify the meaning of the NP.

พงษ์ชอบใส่เสื้อตัวโปรด	amod**(เสื้อ, ตัวโปรด)**
Pong likes to wear his favorite shirt	amod(shirt, favorite)

appos: appositional modifier
An appositional modifier of an NP is an NP immediately to the right of the first NP that serves to define or modify that NP. It includes parenthesized examples.

พณฯ สมัคร นายกรัฐมนตรี	appos**(สมัคร, นายกรัฐมนตรี)**
His Excellency Samak, the Prime Minister,	appos(Samak, Prime Minister)

aux: auxiliary
An auxiliary of a clause is a non-main verb of the clause, e.g. pre- and post-verb auxiliaries.

ฉันจะไปอเมริกาเดือนหน้า	aux(**ไป, จะ**)
I will travel to the US next month.	aux(travel, will)

auxpass: passive auxiliary
A passive auxiliary of a clause is a non-main verb of the clause which contains the passive information

เขาถูกพวกผู้ก่อการร้ายจับตัวไป	auxpass(**จับตัว, ถูก**)
He was kidnapped by terrorists	auxpass(kidnapped, was)

classifier: Classifier
A classifier is a word that is used to classify the referent of a noun according to its meaning and is commonly used in counting.

เขากินข้าวตั้งหนึ่งหม้อ	classifier(**ข้าว, หม้อ**)
He ate one whole pot of rice.	classifier(rice, pot)

cc: coordination
A coordination is the relation between an element of a conjunct and the coordinating conjunction word of the conjunct.

ฉันจะไปซื้อกาแฟกะน้ำตาล	cc(**กาแฟ, กะ**)
I am going to buy some coffee and sugar	cc(coffee, and)

ccomp: clausal complement
A clausal complement of a verb is a dependent clause with an internal subject which functions like an object of the verb.

ฉันไม่ชอบคนไม่ทำงาน	ccomp(**ชอบ, ทำงาน**)
I don't like people who don't work	ccomp(like, work)

compln: complementizer
A complementizer of a clausal complement is the word introducing it. It will be the subordinating conjunction

เขาบอกฉันว่าเขาป่วย	compln(**บอก, ว่า**)
He told me that he is sick	compln(told, that)

conj: conjunct
A conjunct is the asymmetric relation between two elements connected by a coordinating conjunction. The head of the relation is the first conjunct and other conjunctions depend on it via the *conj* relation.

ฉันจะไปซื้อกาแฟกะน้ำตาล	cc(**กาแฟ, น้ำตาล**)
I am going to buy some coffee and sugar	cc(coffee, sugar)

csubj: clausal subject
A clausal subject is a clausal syntactic subject of a clause, i.e. the subject is itself a clause. The governor of this relation might not always be a verb.

ใครวิ่งชนะจะได้รางวัล	csubj(**ได้, วิ่ง**)
Whoever wins the race will get a prize.	csubj(get, wins)

csubjpass: clausal passive subject
A clausal passive subject is a clausal syntactic subject of a passive clause.

เด็กอ่อนแอมักถูกเพื่อนแกล้ง	csubjpass(**แกล้ง, อ่อนแอ**)
A child who is weak often gets bullied by his peers.	csubj(bullied, weak)

det: determiner
A determiner is the relation between the head of an NP and its determiner.

เสื้อนี่สวย	det**(เสื้อ, นี่)**
The shirt is beautiful.	det(shirt, the)

dobj: direct object
The direct object of a VP is the noun phrase which is the accusative object of the verb; the direct object of a clause is the direct object of the VP which is the predicate of that clause.

ครูกำลังบอกคะแนนนักเรียน	dobj**(บอก, คะแนน)**
The teacher is telling students the grades.	dobj(telling, grades)

expl: expletive
An expletive is a word considered as regularly filling the syntactic position of another. This relation captures an existential 'there' and the subject 'it'.

มันเป็นเรื่องดีที่ทุกคนจะอดออม	expl**(เป็น, มัน)**
It is a good thing that everybody saves.	expl(is, it)

iobj: indirect object
The indirect object of a VP is the noun phrase which is the dative object of the verb; the indirect object of a clause is the indirect object of the VP which is the predicate of that clause.

ครูกำลังบอกคะแนนนักเรียน	iobj**(บอก, นักเรียน)**
The teacher is telling students the grades.	iobj(telling, students)

mark: marker
A marker of an adverbial clausal complement (*advcl*) is the word introducing it. It will be a subordinating conjunction different from : e.g.

ถ้าฝนตกเขาจะไม่ออกไปข้างนอก	advcl**(ตก, ถ้า)**
If it rains, he will not go outside.	advcl(rains, if)

neg: negation modifier
The negation modifier is the relation between a negation word and the word it modifies.

ถ้าฝนตกเขาจะไม่ออกไปข้างนอก	advcl**(ออก, ไม่)**
If it rains, he will not go outside.	advcl(go, not)

nn: noun compound modifier
A noun compound modifier of an NP is any noun that serves to modify the head noun.

ราคาน้ำมัน	nn**(ราคา, น้ำมัน)**
Oil price	nn(price,oil)

nsubj: nominal subject
A nominal subject is a noun phrase which is the syntactic subject of a clause. The governor of this relation might not always be a verb.

พวกผู้ก่อการร้ายจับตัวเขาไป	nsubj**(จับตัว, ผู้ก่อการร้าย)**
The terrorists kidnapped him.	nsubj(kidnapped, terrorists)

nsubjpass: passive nominal subject
A passive nominal subject is a noun phrase which is the syntactic subject of a passive clause.

เขาถูกพวกผู้ก่อการร้ายจับตัวไป	nsubjpass**(จับตัว, เขา)**
He was kidnapped by terrorists.	nsubjpass(kidnapped, he)

num: numeric modifier
A numeric modifier of an NP is any number phrase that serves to modify the meaning of the NP.

เขากินข้าวตั้งหนึ่งหม้อ	num**(ข้าว, หนึ่ง)**
He ate one whole pot of rice.	num(rice, one)

pcomp: prepositional complement
The prepositional complement of a preposition is the head of a clause following the preposition.

เขาเพิ่งทราบเกี่ยวกับเรื่องที่คุณป่วย	pcomp**(เกี่ยวกับ, เรื่อง)**
He has just learned about the fact that you are sick.	pcomp(about, fact)

pobj: object of a preposition
The object of a preposition is the head of a noun phrase following the preposition. The preposition in turn may be modifying a noun, verb, etc.)

ที่บ้านเด็กเยอะ	pobj**(ที่, บ้าน)**
At home, there are many children.	pobj(at, home)

poss: possessive modifier
The possession modifier relation holds between the head of an NP and its possessive marker or the modifying noun.

บ้านของเขา	poss**(บ้าน, ของ)**
The house of his.	poss(house, of)

prep/prepc: prepositional modifier
A prepositional modifier of a verb or noun is any prepositional phrase that serves to modify the meaning of the verb or noun.

ที่บ้านเด็กเยอะ	prep**(เด็ก, ที่)**
At home, there are many children.	prep(children, at)

prt: phrasal verb particle
The phrasal verb particle relation identifies a phrasal verb, and hold between the verb and its particle (post-verb auxiliaries or prepositions).

เขาผอมลงกว่าเดิมมาก	prt **(ผอม, ลง)**
He thins down a lot more than before	prt(thin,down)

punct: punctuation
This is used for any piece of punctuation in a clause, if punctuation is being retained in the typed dependencies.

เหนื่อย !	punct**(เหนื่อย, !)**
Tired!	punct(tired,!)

quantmod: quantifier phrase modifier
A quantifier phrase modifier is an element modifying the head of a QP constituent.

เขากินข้าวตั้งหนึ่งหม้อ	quantmod**(หม้อ, ตั้ง)**
He ate one whole pot of rice.	quantmod(pot, whole)

rcmod: relative clause modifier
A relative clause modifier of an NP is a relative clause modifying the NP. The relation points from the head noun of the NP to the head of the relative clause, normally a verb.

เขาเพิ่งทราบเกี่ยวกับเรื่องที่คุณป่วย	rcmod **(เรื่อง, ป่วย)**
He has just learned about the fact that you are sick.	rcmod(fact, sick)

ref: referent
A referent of the head of an NP is the relative word introducing the relative clause modifying the NP.

เขาเพิ่งทราบเกี่ยวกับเรื่องที่คุณป่วย	ref **(เรื่อง, ที่)**
He has just learned about the fact that you are sick.	ref(fact, that)

tmod: temporal modifier
A temporal modifier of a VP or an ADJP is any constituent that serves to modify the meaning of the VP of the ADJP by specifying a time; a temporal modifier of a clause is a temporal modifier of the VP which is the predicate of that clause.

ฉันจะไปอเมริกาเดือนหน้า	tmod(ไป, เดือน)
I will travel to the US next month.	tmod(travel, month)

3 Conclusion

The dependency approach has proven to be suitable for the analysis of many languages including Thai. The analysis presented in this paper represents an initial effort aimed at a wide-coverage description of Thai based on dependency grammar. The list of dependency relations posited is by no means complete and exhaustive. As language usage continually evolves due to many practical applications, the list is bound to increase. We hope that this effort is the step in the right direction.

References

Aroonmanakun, W. 1990. A Dependency Analysis of Thai Sentences for a Computerized Parsing System. *Master's Thesis, Department of Linguistics, Faculty of Arts, Chulalongkorn University (Thailand).*

Potisuk, S. and Harper, M. P. 1996. CDG: An Alternative Formalism for Parsing Written and Spoken Thai. *Proceedings of the Fourth International Symposium on Languages and Linguistics*, 1177-1196.

Sudprasert, S. 2008. A Manual for Annotating Dependency Trees For Thai. Available at the website for The Specialty Research Unit of Natural Language Processing and Intelligent Information System Technology, Department of Computer Engineering, Kasetsart university, Bangkok, Thailand.

Sornlertlamvanich, V., Charoenporn, T., Isahara, H. 1999. Building A Thai Part-of-speech Tagged Corpus (ORCHID). Available at the website for the Linguistics and Knowledge Science Laboratory, National Electronics and Computer Technology Center, Bangkok, Thailand (ftp://www.links.nectec.or.th/pub/paper/virach/orchid.html).

de Marneffe, M. and Manning, C. 2008. The Stanford Typed Dependencies Representation. *Proceedings of the workshop on Cross-framework and Cross-domain Parser*, 1-8.