5

Syntactic Parser for Tamil

K. Saravanan, Ranjani Parthasarathi, T.V. Geetha

Resource Centre for Indian Language Technology Solutions – Tamil,
School of Computer Science and Engineering,
Anna University, Chennai.
(ksaranme@yahoo.com, rp@annauniv.edu, tvgeedir@annauniv.edu)

Abstract

Parser is a main tool in natural language processing like language translation process, grammar checker and multilingual information extraction. It is used to identify the syntactic constituents of a sentence as a tree structure. Parser for Tamil language is a complex process because of the free word order feature of the language due to the tight coupling between the morpholocial and syntactic levels. This paper decribes a parser of Tamil language which handles simple sentences and complex sentences with clauses.

Introduction

In this paper the implementation of a Tamil parser is discussed. The parser is a process, which identifies syntactic constituents of a sentence and represents the same using a parse tree. Parsing is a complex process due to the ambiguity present at the morphological and syntactic level. Most parsing in natural language is context dependent. Many parsers have been built for English language [1][2]. Mechanism used to handle English parsing is based on its fixed order nature. In this paper a parser is described which handles Tamil, a predominantly free word order language.

Feature of Tamil Language

Tamil is predominantly a free word order language. In Tamil sentence verbs themselves are options. However if a verb component occurs, it normally occurs in the final position of the sentence. Generally Tamil sentence follows the subject, object, and verb pattern. However the interchange of subject, object is acceptable.

Note that TF refers translitrated form and EF refers English form. The Tamil English transliteration scheme is given in figure 1.

Ex. 1. இராமன் சென்னைக்கு வந்தான்

TF: iraaman cennaikku van'taan

EF: Raman Chennai came

2. சென்னைக்கு இராமன் வந்தான்

TF: iraaman cennaikku van'taan

EF: Chennai Raman came

In this example the first sentence is of the form subject, object and verb, whereas the second sentence is of the form object, subject and verb.

This free word nature of Tamil language is made possible by it being a morphologically rich language. In fixed word order language English the position of the word plays an important part in determining the syntactic function of the word. In particular, free word order language like Hindi local word groups help in determining syntactic function.

Tamil language morphology contributes to the free word order nature of the language in two ways. Case markers indicating thematic cases like subject, object etc. are conveyed in English by either the position of the noun or the preposition associated with the noun. In Hindi thematic cases are generally indicated by preposition. In Tamil thematic cases are indicated by case suffixes attached to the noun itself. This means that the case suffixes attached nouns can occur any where in the sentence. But its thematic case can be determined in most cases by its suffix.

Ex. In the above example sentences, the subject 'இராமன்' (TF: iraaman, EF: Raman) can be identified by the noun with nominative case and the object 'சென்னைக்கு°' (TF: cennaikku, EF: Chenna) can be identified by the dative case marker 'க்கு°'(TF: ikku).

Consider the equivalent English sentence "Raman came to Chennai".

In this sentence the object 'Chennai' can be identified by the preposition 'to' and the subject 'Raman' can be identified by its position.

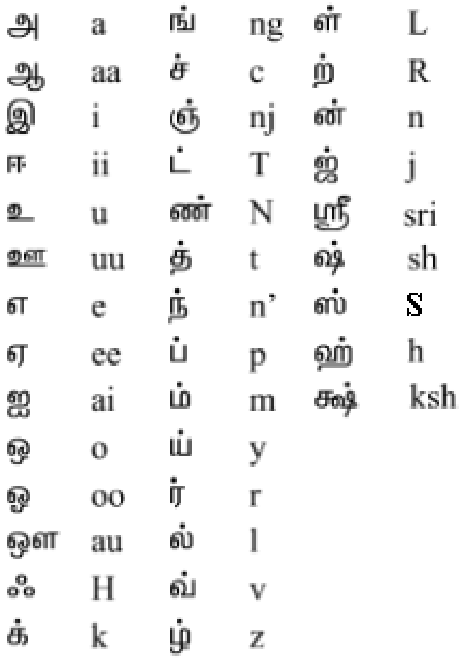
The second important positional grouping of words in fixed word languages occur when words indicating tense, aspect and mood have to be in the local context of the main verb. However in Tamil, auxiliaries indicating tense, aspect and mood are attached to the main verb as suffixes. This eliminates the need for local word grouping pertaining to verbs. Thus the use of suffixes for case and tense, aspect and mood of verb allow Tamil to be predominantly a free word order language. Another aspect of Tamil language as in the case of Hindi is the need to have person, gender and number agreement between subject and verb of the sentence.

Ex. In the example sentence the subject 'இராமன்' (TF: iraaman, EF: Raman) is third person, singular and masculine gender. The verb also contains the suffix indicating third person, singular and masculine.

These three features of Tamil language make the approach of parsing Tamil language different. The symbol table given below explains the symbols used.

Symbol	Description
S	Sentence
NC	Noun constituent
VC	Verb constituent
N	Noun
V	Verb
Adj	Adjective
Adv	Adverb
Vpl	Verbal participle
Rpl	Relative participle
Con	Connective
NNC	Noun clause
ADJC	Adjective clause
ADVC	Adverb clause

Table 1. Symbols and their description



அ	a	ங்	ng	ள்	L
ஆ	aa	ச்	c	ற்	R
இ	i	ஞ்	nj	ன்	n
ஈ	ii	ட்	T	ஜ்	j
உ	u	ண்	N	ஸ்ரீ	sri
ஊ	uu	த்	t	ஷ்	sh
எ	e	ந்	n’	ஸ்	S
ஏ	ee	ப்	p	ஹ்	h
ஐ	ai	ம்	m	க்ஷ்	ksh
ஒ	o	ய்	y		
ஓ	oo	ர்	r		
ஔ	au	ல்	l		
ஃ	H	வ்	v		
க்	k	ழ்	z		

Fig.1. Tamil English transliteration scheme.

Grouping

In fixed word order language like English a noun constituent can be attached with the verb constituent or with another noun constituent at any level of the tree except the root and leaf level. This means that English has a multilevel phrasal structure tree as shown in fugure2.

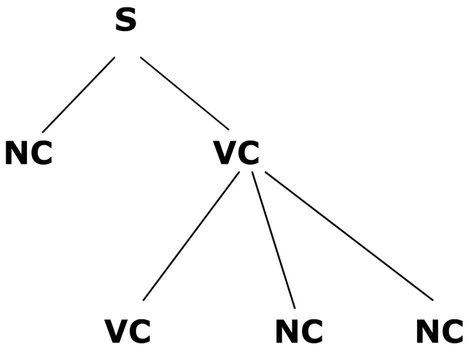


Fig. 2. Sample tree for English

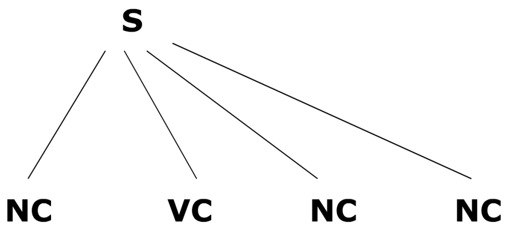


Fig. 3. Sample tree for Tamil

In Tamil the two main components of the parse tree are noun constituents(NC) and verb constituents (VC). The noun constituent has the form.

1. (adjective)* (adjective clause)* (adjective)* (adjective clause)* (adjective)* noun (associated case marker) (post position)

Ex. அழகிய நான் வேடந்தாங்கலில் பார்த்த நீண்ட கால்களை கொண்ட வெண்ணிற இடைவிடாது சைபீரியாவிலிருந்து வந்த சிறிய பறவையைப் போல

TF : azakiya n'aan veeTan'taangkalil paartta n'iiNTa kaalkaLai koNTa veNNiRa iTaiviTaatu caipiiriyaavilirun'tu van'ta ciriya paRavaiyaip pool

EF : beautiful I Vedanthangal saw long legs having white contineous Syberia came small bird like

(or)

2. noun clause

Ex. இராமன் நன்றாகப் பாடுவான் என்று

TF : iraaman n'anRaakap paaTuvaan enRu

EF : Raman well sing that

Here the braces indicates optional and ‘*’ indicates multiple occurrence is possible. The clauses are generally indicated by special cue suffixes / cue phrases. The verb constituent has the form

(adverb clause)* (adverb)* verb (suffix)*

Ex. கோவையிலிருந்து சென்னைக்கு வந்து தமிழக அமைச்சர்களுடன் ஆலோசித்து மெதுவாக மற்றும் தெளிவாக பேசிக்கொண்டிருந்தார்

TF: koovaiyilirun'tu cennaikku van'tu tamizaka amaiccarkaLuTan aaloocittu metuvaaka maRRum teLivaaka peecikkoNTirun'taar

EF: Kovai Chennai came Tamil Ministers discuss slowly and clearly was-speaking

The sentence is of the form NC* (VC) (or) (NC)* VC

In Tamil, since noun constituents contains case markers in general and there is no pre-position phase in Tamil a noun constituent can be attached directly with the root of the tree [3] as shown in the figure 3.

In the simple sentences two types of grouping are possible. One is grouping adverbs with the main verb and the other is grouping adjectives with the noun. Words are grouped based on the function they represent. In general the adjectives occur adjacent to the noun, which they represent. But since Tamil is a free word language adverbs can occur anywhere in the sentence prior to verb. It means that some other nouns can also be present between the adverbs and verb. Since words are grouped irrespective of the position, in the output of the parser their position also has to be given.

Ex. Consider the Tamil sentences.

3. அழகான பறவை பசுமையான சோலையில் வேகமாக பறந்தது

TF: azakaana paRavai pacumaiyaana coolaiyil veekamaaka paRan'tatu

EF: beautiful bird greenish garden fast flew

4. அழகான பறவை வேகமாக பசுமையான சோலையில் பறந்தது

TF: azakaana paRavai veekamaaka pacumaiyaana coolaiyil paRan'tatu

EF: beautiful bird fast greenish garden flew

In this example the adjectives occur adjacent to its noun in both the sentences, whereas the adverb in the first sentence occurs adjacent to its verb and in the second sentence it occurs prior to the verb with other words in-between.

In the sentences with clauses, three types of grouping are possible. They are adverb clause with the corresponding verb to form the verb constituent, adjective clause with the corresponding noun to form noun constituent and noun clause which alone forms the noun constituent. As explained earlier the clausal portion are grouped with its corresponding words on the basis of their functional representation. The adjective clauses and its corresponding noun are adjacent to each other but the adverb clause can occur any where in the sentence prior to its corresponding verb. That is some noun constituents can also be present between the adverb clauses and its corresponding verb.

Ex. 5. இராமன் நன்றாகப் பாடுவான் என்று வகுப்பிலேயே முதலாவதாக வந்த பாலுவிற்கு அவனுடைய பாடலைக் கேட்டு தெரிந்தது

TF: iraaman n'anRaakap paaTuvaan enRu vakuppileeyee mutalaavataaka van'ta paaluviRku avanuTaiya paaTalaik keeTTu terin'tatu

EF: Raman well sing that class first come Balu his song hear know

In this sentence,

Noun clause : இராமன் நன்றாகப் பாடுவான் என்று

TF: iraaman n'anRaakap paaTuvaan enRu

EF: Raman well sing that

Adjective clause : வகுப்பிலேயே முதலாவதாக வந்த

TF: vakuppileeyee mutalaavataaka van'ta

EF: class first come

Adverb clause : அவனுடைய பாடலைக் கேட்டு

TF: avanuTaiya paaTalaik keeTTu

EF: his song hear

This sentence can be rearranged so that the adverb clause can occur before somewhere to its verb as follows.

6. இராமன் நன்றாகப் பாடுவான் என்று அவனுடைய பாடலைக் கேட்டு வகுப்பிலேயே முதலாவதாக வந்த பாலுவிற்கு தெரிந்தது

TF: iraaman n'anRaakap paaTuvaan enRu avanuTaiya paaTalaik eeTTu vakuppileeyee mutalaavataaka van'ta paaluviRku terin'tatu

EF: Raman well sing that his song hear class first come Balu know

Discontinuous constituents can be handled by the combination of phrase structure rules and syntactic transformations or scrambling rules [4].

Parsing can be performed only after morphological analysis has been done.

Morphological analyzer

The morphological analyzer is a tool basically used to identify the part of speech tag of a word. It takes a derived word as input and separates it into the root word and the corresponding suffixes. The function of each suffix is also indicated. It uses rule-based approach and has two major modules noun analyzer and verb analyzer. The steps involved are as follows:

1. Given an input string, the morphological analyzer starts scanning the string from right to left to look for suffixes. A list of suffixes is maintained for this purpose.
2. It then, searches for the longest match in the suffix list.
3. The morphological analyzer then removes the last suffix, determines its tag and adds it with the word’s suffix list.
4. It then checks the remaining part of the word in the dictionary and exits if the entry found.
5. According to the identified suffix, it generates the next possible suffix list.
6. It repeats from second step with the current suffix list.

Ex. 1.

வந்துவிட்டான் (TF: van'tuviTTaan)

வா (TF: va)< Verb >

ந்து (TF: n’tu)< Verbal Participle >

விடு (TF: viTu)< Auxiliary Verb >

ட் (TF: T)< Past Tense Marker >

ஆன் (TF: aan)< Third Person Singular Mas. >

Ex. 2.

வகுப்பில் (TF: vakuppil)

வகுப்பு (TF: vakuppu)< Noun >

இல் (TF: il)< LocCase >

Parsing

Morphologically analyzed sentence is input to the actual parsing process. The morphological analyzer cannot analyze all words. This may be due to the fact that the particular word does not present in the dictionary or the word may be associated with more than one syntactic category. These unidentified words can be identified using linguistic based heuristic rules.

Ex. 7. இராமன் நன்றாகப் பாடுவான்

TF : iraaman n'anRaakap paaTuvaan

EF : Raman well sing

In this sentence if the verb dictionary doesn’t contains the verb ‘பாடுவான்’ (TF: paaTuvaan, EF: sing), it can be identified as a verb by the heuristic rule that it is the last word of the sentence and its preceding word is an adverb.

Simple sentence

For the simple sentence, adjectives and adverbs are attached with their corresponding words irrespective of the presence of adverb anywhere in the sentence prior to its verb. Then the NC (noun constituent) and VC (verb constituent) are attached with the root of the tree.

Ex. Consider the above example sentences 4 & 5.

The tree structure of these two sentences given by the parser is shown below.

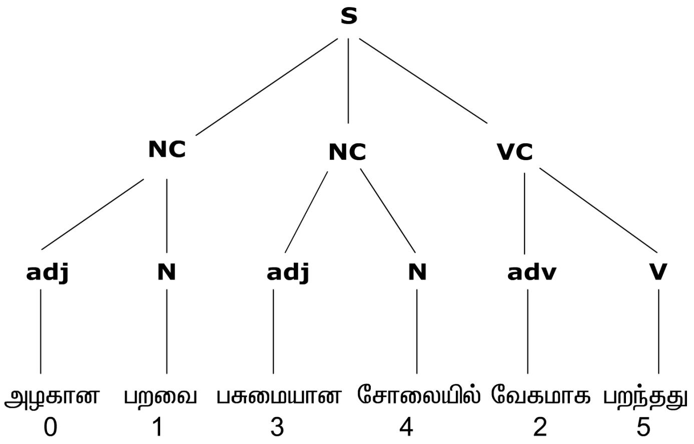


Fig. 4. Parser tree for example sentence 3

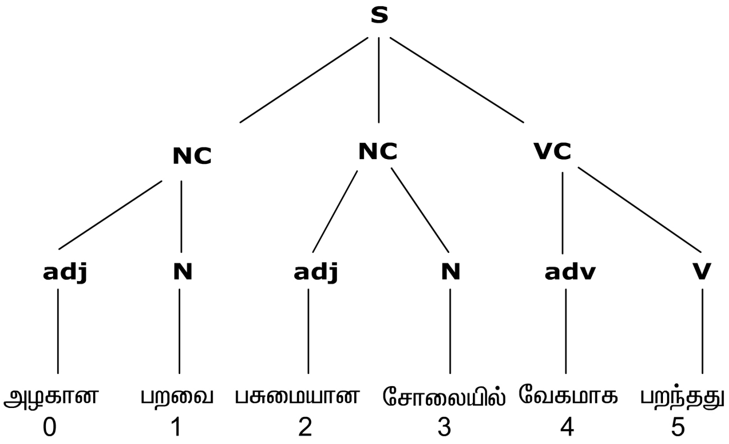


Fig. 5. Parser tree for example sentence 4

Note that these two tree structures are same except for the word positions.

PNG marker of the verb performs an important role in identifying the subject of a sentence. A noun is a subject if it first matches with the verb of the sentence in the following three ways while processing from left to right.

i. Person of the noun must be same as the person of the verb.

ii. Number of the noun must be same as the number of the verb.

iii. Gender of the noun must be same as the gender of the verb.

The third condition is simple in the case of pronouns. But for other words the problem is not trivial. In order to perform subject classification the gender, animate and inanimate informa-tion of a noun is required. Thus semantic information may have to be associated with noun words in the dictionary.

Ex. 8. இராமன் படம் பார்த்தான்

TF: iraaman paTam paarttaan

EF: Raman picture saw

In this sentence the verb contains third person, singular and masculine gender. At present the information about the two nouns are third person and singular. In this example Roman and padam are both nominative and the position does not indicate subject in Tamil. Here information about Raman and fruit are required to identify subject and object.

Complex Sentences

Complex sentences with noun clause, adjective clause and adverb clause are considered. These complex sentences are parsed by the following two steps.

1. Conversion of complex sentence into simple sentence by grouping the three clauses with their corresponding words and forming NC and VC.
2. Parsing the simple sentence can be done as mentioned earlier.

Grouping the clauses is the main process of the parser. The clauses are generally indicated by special cue suffixes / cue phrases. Grouping is done by the position of the cues and linguistic based heuristic rules.

Ex. Consider the sentence with all the three clauses.

5. இராமன் நன்றாகப் பாடுவான் என்று வகுப்பிலேயே முதலாவதாக வந்த பாலுவிற்கு அவனுடைய பாடலைக் கேட்டு தெரிந்தது

TF: iraaman n'anRaakap paaTuvaan enRu vakuppileeyee mutalaavataaka van'ta paaluviRku avanuTaiya paaTalaik keeTTu terin'tatu

EF: Raman well sing that class first come Balu his song hear know

After step 1:

NC *பாலுவிற்கு* (TF: paaluviRku, EF: Balu)(with its adjective clause) *தெரிந்தது* (TF: terin’tatu)(with its adverb clause)

After step 2:

The result of the step 1 will give the tree structure as shown below in figure 6.

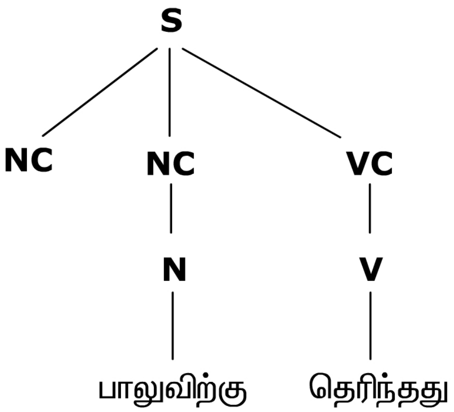


Fig. 6. Parse tree for converted sentence

This simple tree can be expanded with the three clauses as shown below.

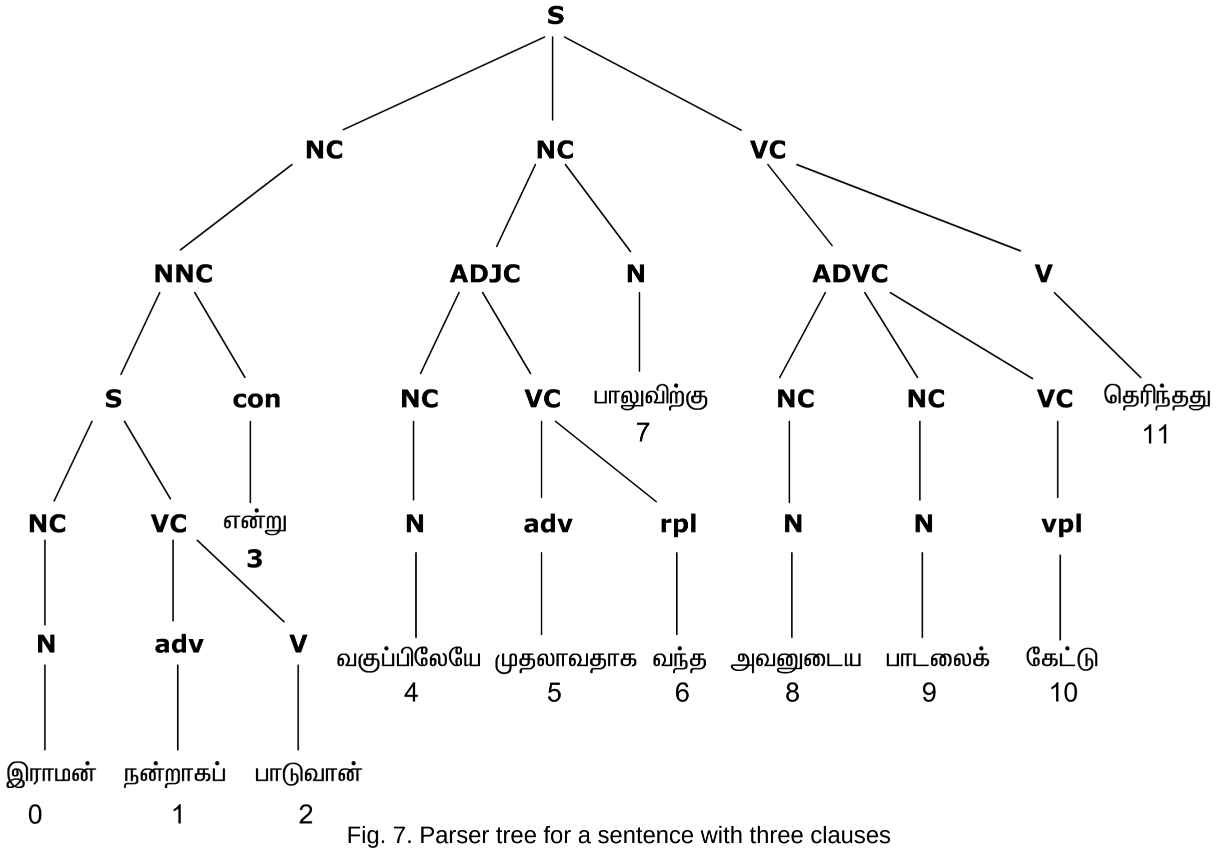


Fig. 7. Parser tree for a sentence with three clauses

The tree structure of the example sentence 6 is also similar to the above tree structure except the word position. The parser takes care of positional mixing of adverbial clauses. In a complex sentence subject can be identified in the converted simple sentence as mentioned earlier.

Complex sentences with more than one verb with conjunctions can be parsed by the following steps.

i. Resolve the sentence into more than one simple sentence by eliminating the conjunctions.

ii. If any one of this simple sentence doesn't have the subject, consider the subject of the previous sentence.

iii. Parse these simple sentences as sub sentences of the main sentence separately and join these parsed sub sentences with conjunctions to form the syntactic constituents of the main sentence.

Ex. 9. நான் பாடம் படித்தேன் மேலும் அவன் படம் பார்த்தான்

TF: n'aan paaTam paTitteen meelum avan paTam paarttaan

EF: I read lesson and he saw picture

This example sentence contains two verbs and a conjunction. If this sentence is divided into two sentences by eliminating the conjunction 'மேலும்' (TF: meelum, EF: and), there exists a noun 'அவன்' (TF: avan, EF: he) which matches with the second verb 'பார் த் தான்' (TF: paarttaan, EF: saw) and so act as a noun of the second sentence. So these two sentences can be parsed separately and joined with the conjunction as sub sentences of the main sentence.

10. நான் படம் பார்த்தேன் மேலும் பாடம் படித்தேன்

TF: n'aan paaTam paTitteen meelum paTam paartteen

EF: I read lesson and saw picture

This sentence also contains two verbs and a conjunction. If this is divided, the second part will not have the subject. So the subject of the first sentence can be considered as its subject and can do the further process.

Conclusion

The parser handles simple sentences and also complex sentence with multiple noun, adjective and adverb clauses. Handling of conjunctions has been tackled to a limited extent.

Subject identification requires gender, animate and inanimate information of a noun while checking the noun and verb coordination. This problem is yet to be tackled.

The addition of rules for semantic dependencies can enhance the performance of the parser. The parser needs to handle and provide more than one parse tree for syntactically ambiguous sentences. In addition there is a need to handle and/or correct syntactically incorrect sentences.

Reference:

1. Michael A. Covington, "A Dependency parser for free and fixed order Languages", University of Georgia Athens.
2. J. Woch, F. Widmann, "Implementation of schema-TAG-parser", University of Koblenz-Landau.
3. Thomas Lehmann, "A Grammar of Modern Tamil", Pondichery Institute of Linguistics and Culture
4. Ross J.R., "Constraints on variables in syntax", Dissertation, 1967.