

**A DEEP SYNTACTIC PARSER
FOR THE TAMIL LANGUAGE**

Kengatharaiyer Sarveswaran

178097E

Degree of Doctor of Philosophy

Department of Computer Science & Engineering

University of Moratuwa
Sri Lanka

September 2022

**A DEEP SYNTACTIC PARSER
FOR THE TAMIL LANGUAGE**

Kengatharaiyer Sarveswaran

178097E

Thesis submitted in partial fulfillment of the requirements for the
degree of Doctor of Philosophy

Department of Computer Science & Engineering

University of Moratuwa
Sri Lanka

September 2022

DECLARATION

I declare that this is my own work and this dissertation does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature: ***UOM Verified Signature***

Date: 20/09/2022

The above candidate has carried out research for the PhD thesis under our supervision.

Name: Professor Gihan Dias, University of Moratuwa, Sri Lanka.

Signature of the Supervisor: ***UOM Verified Signature***

Name: Professor Miriam Butt, University of Konstanz, Germany.

Signature of the Supervisor: ***UOM Verified Signature*** Date: 20/09/2022

ABSTRACT

A Deep Syntactic Parser for the Tamil Language

Natural Language Processing (NLP) applications have become integral to human life. A syntactic parser is a vital linguistic tool that shows syntactic relations between the words in a sentence. These may then be mapped to a tree, a graph, or a formal structure. Syntactic parsers are helpful for building other NLP applications. In addition, they help linguists to understand a language better and perform cross-lingual linguistic analysis. A syntactic parser that performs a deeper analysis and captures argumentative, attributive and coordinative relations between the words of a given sentence is called a deep syntactic parser. Tamil is considered a low-resourced language in terms of tools, applications, and resources available for others to use and build NLP applications or carry out linguistic analyses. Not many resources, such as treebanks and annotated corpora, or linguistic analysis tools such as POS taggers or morphological analysers, are publicly available for Tamil. Available off-the-shelf language-agnostic syntactic parsers show comparatively low performance because of the rich morphosyntactic properties of Tamil. This study elaborates on how I developed the first grammar-driven parser for Tamil, which uses the Lexical-Functional Grammar formalism, and a state-of-the-art data-driven parser using the Universal Dependencies framework. I have also proposed an approach to evaluate a syntactic parser's syntactical coverage, experimented with transition-based and graph-based approaches, and for the first time, tried multi-lingual training to develop a data-driven parser for Tamil. A part of speech tagger, a morphological analyser cum generator, pre-processing tools, and treebanks are the other tools and resources I have developed to facilitate the development of the parsers. While all these tools give the current best score for their respective tasks, these resources are also available online for others to build upon. Moreover, the study also documents my contributions toward understanding different linguistic aspects of the Tamil language.

Keywords: Deep Syntactic Parser; Grammar-driven parser; Data-driven parser; Part of Speech tagger; Morphological Analyser

DEDICATION

அப்பா - அம்மா

appā - ammā

‘Father - Mother’

for their unconditional love, support, and being the reason of who I am today.

பெரியப்பா - பெரியம்மா

periyappā - periyammā

‘Uncle - Aunt’

for being my guardians when crossing the most important part of my life.

இயற்கை

iyarkai

‘the great Nature’

(the god)

for always putting together and aligning things I required for the progressions of this study.

ACKNOWLEDGEMENTS

I am very grateful to the people who have supported me in various ways since the beginning of this study in 2018 to complete this study.

First, I must thank my two perfect supervisors, Professor Gihan Dias and Professor Miriam Butt. I would not have completed this study without their tremendous support. Their detailed and insightful comments have significantly improved my research work. I am additionally thankful to them for creating valuable opportunities to widen my knowledge and academic network.

I thank Professor Gihan Dias for encouraging me to start my PhD research and offering me the wonderful opportunity to be his doctoral student. He always knew the right moment when to lend me a word of encouragement and give me a push. I am grateful for that and for the valuable guidance he has provided to complete this study. Professor Gihan Dias also supported me with the funding to carry out research, attend conferences, and organise academic events.

I thank Professor Miriam Butt, my other supervisor, for helping me with the computational linguistics part of my study. She hosted me twice as a visiting researcher at the University of Konstanz and created opportunities to meet other leading scholars in this field of study. Professor Miriam also provided me with financial support to be able to attend conferences and supported me in co-organising workshops and a summer school. I am grateful for all these and the trust she kept in me.

I want to express my gratitude to the progress evaluation panel members, Professor Subathini Ramesh and Dr Charith Chitraranjan, and research coordinators Professor Sanath Jayasena and Dr Shehan Fernando, for their continuous input and guidance in tailoring my research.

I am grateful to the following thesis evaluation panel members for their constructive comments and feedback to improve the thesis and plan my future work:

- Professor Jagath Premachandra - Professor at the University of Moratuwa, Sri Lanka.
- Professor Mary Dalrymple - Professor of Syntax at the University of Oxford, United Kingdom.
- Professor Sarmad Hussain - Professor of Computer Science, University of Engineering and Technology, Pakistan.

- Dr Uthayasanker Thayasivam - Senior lecturer in Computer Science at the University of Moratuwa, Sri Lanka.
- Dr Amal Shehan Perera - Senior lecturer in Computer Science at the University of Moratuwa, Sri Lanka.

I should thank my friend Dr Maris Camilleri for helping me with language editing and providing feedback on my write-up. I know, at times, she sacrificed her personal and family time to do these. I am very grateful to her in this respect.

I would like to thank my colleagues at the Department of Computer Science, the University of Jaffna for encouraging me to start the PhD research, and the administration of the University of Jaffna for providing me with the required permission and leave to carry out the research.

I am also thankful to the wonderful staff and researchers of the National Languages Processing centre, and the head and the staff of the Department of Computer Science and Engineering at the University of Moratuwa for their support, input, and the provision of all required resources. Moreover, I am grateful to them for making my time at the University of Moratuwa memorable and valuable. I would also like to thank the director and the staff Faculty of Graduate Studies for facilitating all administrative matters related to the registration and evaluation of my study.

I also thank my friends in the Computational Linguistics group at the University of Konstanz for making my stay at Konstanz productive and memorable. I would especially like to thank Jessica Zipf for her support in solving issues related to XLE.

I am thankful to Paul Meurer at the University of Bergen for providing the required permission and helping to set up the Tamil grammar on INESS. I am grateful to Dr Dan Zeman, Charles University, the Czech Republic, for helping me sort out technical issues and set up a Tamil Universal Dependencies treebank.

I would like to thank Professor Lauri Karttunen from Stanford University for introducing me Foma (a Finite-State compiler), which cleared one of my greatest obstacles in the development of the morphological analyser, and Dr Mans Hulden from the University of Colorado Boulder, for helping me sort out issues related to Foma.

I thank Dr Parameswari Krishnamurthy from the University of Hyderabad, for working with me to create a Tamil treebank that I then used as a part of my test set. I also would like to thank Professor Rajendran Sankaravelayuthan, Amrita Vishwa Vidyapeetham, Dr S. Rajamathangi, Jawaharlal Nehru University, and Ms Sujiththa Srikantharajah, University of Jaffna for helping me with reviewing linguistic annotations.

I owe a lot to my wife, Sharanyaa and daughters, Thenmozhi and Poongkuzhalii, for their patience and love. It would have been much easier for them if I had continued working at the University of Jaffna, but they encouraged me to go ahead with my studies. Our little daughter Nilavezhilii is the best thing that happened to us during

this challenging time. I know they made sacrifices to help me stay focused on my studies. I also thank my parents and brothers for supporting me in everything with selfless generosity.

காலத்தி னாற்செய்த நன்றி சிறிதெனினும்
ஞாலத்தின் மாணப் பெரிது. - திருக்குறள் (102)

kālatti nārceyta nanri ciriteninum
ñālattin māṇap peritu. - tirukkural (102)

“A favour conferred in the time of need,
though it be small (in itself),
is (in value) much larger than the world.”

Thank you!

TABLE OF CONTENTS

Declaration	i
Abstract	ii
Dedication	iii
Acknowledgements	iv
Table of content	vii
List of Figures	xii
List of Tables	xiv
List of abbreviations	xvi
Transliteration schema	xx
1 Introduction	1
1.1 The Problem	2
1.1.1 The need for a syntactic parser	2
1.1.2 Constraints and Challenges	3
1.2 Research objectives	4
1.2.1 Scope	5
1.2.2 Deepness of my parsers	6
1.3 The Tamil language	6
1.3.1 Tamil Alphabet	7
1.3.2 Tamil Grammars	7
1.3.3 Tamil and other Dravidian languages	9
1.3.4 Encoding Tamil letters	10
1.3.5 Institutions working on Tamil language computing	10
1.4 Thesis outline	11
2 Part of Speech Tagging	14

2.1	Literature survey	14
2.1.1	POS Tagsets	14
2.1.2	POS Taggers	15
2.1.3	Datasets	17
2.2	<i>Thamizhi</i> POStagger	18
2.2.1	The POS tagset	18
2.2.2	The approach	20
2.3	Methodology	20
2.3.1	Data Preparation	21
2.3.2	Training	24
2.3.3	Evaluation and Discussion	25
2.4	Conclusion and future work	27
2.4.1	Conclusion	27
2.4.2	Future work	28
3	Morphological Analysis and Generation	29
3.1	Introduction	29
3.2	Tamil Morphology	29
3.2.1	Nominal Morphology	30
3.2.2	Verbal Morphology	34
3.2.3	Adjectival Morphology	39
3.2.4	Morphology of adverbs	39
3.3	The Finite-State approach	40
3.3.1	Finite-State Morphology	41
3.3.2	Foma	42
3.4	Literature survey	43
3.4.1	Approaches for developing Morphological analysers	43
3.4.2	Morphological analysers for South Asian languages	44
3.4.3	Tamil morphological analysers	44
3.5	Development of <i>Thamizhi</i> Morph	46
3.5.1	The approach for developing <i>Thamizhi</i> Morph	46
3.5.2	The technology stack	47
3.5.3	Scope of annotations	48
3.5.4	Morpheme labels	49
3.6	<i>Thamizhi</i> Morph	52
3.6.1	Pre-processing	53
3.6.2	Compilation of Lexicons	53
3.6.3	Meta-Morph rules	54
3.6.4	Orthographical rules	58

3.6.5	Morphological guesser	59
3.7	Evaluation and Discussion	59
3.7.1	Comparing <i>ThamizhiMorph</i> and the IIT Parser	60
3.7.2	Evaluating <i>ThamizhiMorph</i> using UD Tamil Treebank	62
3.8	Morphological generation	63
3.9	<i>ThamizhiMorph</i> to UD tagging	63
3.10	Conclusion	66
3.11	Future work	67
4	Treebanks, Parsing, and Tamil Syntax — an overview	69
4.1	Treebanks and Grammars	69
4.1.1	Grammar formalisms	69
4.1.2	Treebanks	70
4.2	Syntactic parsing: Literature and Approaches	71
4.2.1	Constituency and Dependency parsing	71
4.2.2	Grammar-based and Data-driven parsing approaches	72
4.2.3	Algorithms for dependency parsing	72
4.2.4	Parsing in Tamil	74
4.3	Tamil syntax — an overview	75
4.3.1	Background	75
4.3.2	Nouns and their structures	76
4.3.3	Agreement	82
4.3.4	Negation	84
4.3.5	Postpositions in Tamil	85
4.3.6	Verbs and their syntactic structures	85
4.3.7	Clitics	91
4.3.8	Coordination	92
4.3.9	Interrogatives	92
4.3.10	Complex clauses in Tamil	93
4.4	Summary	95
5	Grammar-driven parsing	96
5.1	Lexical Functional Grammar (LFG)	96
5.1.1	ParGram project	98
5.2	Methodology	99
5.2.1	Data	99
5.2.2	Development Environment	99
5.2.3	Annotation scheme	101
5.2.4	Implementation	101
5.3	The LFG analysis of concepts and syntactic constructions	103

5.3.1	<i>Sandhi</i>	103
5.3.2	Intransitives	104
5.3.3	Ditransitive construction	104
5.3.4	Imperatives	106
5.3.5	Clausal argument structures	107
5.3.6	Oblique arguments in Tamil	110
5.3.7	Interjection, Word-order, and Pro-drop	111
5.3.8	Copula and Null-Copula constructions	113
5.3.9	Dative Subjects	115
5.3.10	Complex Predicates and handling of multiple predicational heads	116
5.3.11	Passives	119
5.3.12	Causatives	122
5.3.13	Coordination	125
5.4	LFG Treebank	126
5.5	Evaluation and Discussion	127
5.6	Conclusion	128
5.7	Future work	129
6	Data-driven parsing	130
6.1	Universal Dependencies (UD)	130
6.2	Levels of annotation in UD	132
6.2.1	POS annotation	132
6.2.2	Morphological annotation	133
6.2.3	Syntactic annotation	133
6.3	Tamil Universal Dependencies Treebanks	135
6.4	Tools for syntactic parsing	136
6.4.1	Multi-lingual learning	137
6.4.2	uuparser	137
6.4.3	Stanza	138
6.4.4	TranKit	138
6.5	Evaluation metrics for data-driven parsers	138
6.6	Design choices	139
6.6.1	Approach	139
6.6.2	Dataset	140
6.7	Methodology	143
6.7.1	Tokenisation	144
6.7.2	Multi-word tokenisation	145
6.7.3	Lemmatisation	145
6.7.4	POS tagging using <i>Thamizhi</i> POST	145

6.7.5	Morphology annotation using <i>ThamizhiMorph</i>	145
6.7.6	Dependency annotation using of <i>ThamizhiUDparser</i>	146
6.8	Analysis of datasets	147
6.8.1	Part of Speech information	148
6.8.2	Morphological features	148
6.8.3	Dependency relations	151
6.9	Training and Evaluation	152
6.10	Error Analysis	154
6.10.1	Nominal predicates	154
6.10.2	Transitives	156
6.10.3	Intransitives	157
6.10.4	Conjoined sentences	158
6.10.5	Complex constructions	161
6.11	Conclusion	162
6.12	Future work	163
7	Conclusion	166
7.1	Software and Resources	168
7.2	Publications	169

LIST OF FIGURES

Figure 1.1	Connection between the key chapters of this thesis.	12
Figure 3.1	Two-level morphology	41
Figure 3.2	Finite-State Transducer network	43
Figure 3.3	How a FST is built using Meta-Morph rules	56
Figure 3.4	A sample output from the IIIT shallow parser	60
Figure 3.5	A sample output from <i>ThamizhiMorph</i>	60
Figure 4.1	A non-projective dependency tree	73
Figure 5.1	XLE architecture	100
Figure 5.2	Analysis of an intransitive construction	104
Figure 5.3	C-structure of a ditransitive construction	105
Figure 5.4	F-structure of a ditransitive construction	106
Figure 5.5	C-structure and f-structure analysis of an imperative	107
Figure 5.6	C-structure of a COMP analysis	108
Figure 5.7	F-structure of a COMP analysis	109
Figure 5.8	C-structure of scrambled COMP analysis	110
Figure 5.9	Analysis of construction with an oblique argument	111
Figure 5.10	An example for an interjection	112
Figure 5.11	Interjection and <i>pro</i> -drop	112
Figure 5.12	Analysis of a null-copula construction	114
Figure 5.13	Analysis of a copula construction	114
Figure 5.14	Analysis of an N+A copula construction	115
Figure 5.15	Analysis of a dative subject construction	116
Figure 5.16	C-structure analysis of a complex predicate	116
Figure 5.17	F-structure analysis of a complex predicate	117
Figure 5.18	C-structure and f-structure analyses of a simple passive	120
Figure 5.19	C-structure analysis of a coordination construction	126
Figure 5.20	F-structure analysis of a coordination construction	126
Figure 6.1	Phases of the Parsing Pipeline	144
Figure 6.2	Dependency analysis using <i>ThamizhiUDparserV2</i>	154
Figure 6.3	Dependency analysis using Stanza	155

Figure 6.4	Dependency analysis of a nominal predicate	155
Figure 6.5	Dependency analysis of the nominal predication	156
Figure 6.6	Dependency analysis of a transitive construction	156
Figure 6.7	Dependency analysis of a transitive construction - Stanza . . .	157
Figure 6.8	Dependency analysis of an N+V predication	157
Figure 6.9	Dependency analysis of the passive construction	158
Figure 6.10	Dependency analysis of a serial verb construction	159
Figure 6.11	Dependency analysis of a coordination construction	160
Figure 6.12	Proposed dependency analysis for a gapping construction . . .	161
Figure 6.13	Dependency analysis of a complex construction	162

LIST OF FIGURES

Table 2.1	Tagsets used for POS tagging of Tamil	15
Table 2.2	Scores of POS taggers for Tamil POS tagging	17
Table 2.3	Available POS tagged datasets for Tamil	17
Table 2.4	UPOS tagset	19
Table 2.6	CoNLL-U formatted text to train the POS tagger	22
Table 2.5	The harmonisation of the Amrita and UPOS tagsets	23
Table 2.7	Training - Development - Testing sets	24
Table 2.8	The evaluation results	27
Table 3.1	Case markers considered for <i>ThamizhiMorph</i>	33
Table 3.2	The Tamil nominal paradigm used for <i>ThamizhiMorph</i>	35
Table 3.3	List of complex verb forms used in <i>ThamizhiMorph</i>	37
Table 3.4	The Tamil Verbal Paradigm used for <i>ThamizhiMorph</i>	38
Table 3.5	Lexically Specified Features	49
Table 3.6	The list of labels used in the current version of <i>ThamizhiMorph</i>	50
Table 3.7	<i>ThamizhiMorph</i> vs. IIIT Tamil Shallow parser	61
Table 3.8	One-to-One mappings between <i>ThamizhiMorph</i> and UD labels	64
Table 3.9	One-to-Many mappings between <i>ThamizhiMorph</i> and UD features	65
Table 5.1	The Tamil language specific morphosyntactic features	102
Table 6.1	CoNLL-U annotation style - used in UD treebanks	132
Table 6.2	The list of dependency relations in UD	134
Table 6.3	Datasets used to implement the syntactic parser	141
Table 6.4	Categorisation of syntactic constructions	142
Table 6.5	<i>ThamizhiUDparser</i> process pipeline	144
Table 6.6	Multi-word tokens handled by the parser	145
Table 6.7	LASs of Multi-lingual models	147
Table 6.8	Fields used in CoNLL-U data format	148
Table 6.9	Part of Speech tags used in the Tamil annotations	148
Table 6.10	Morphological features used in the datasets	149
Table 6.11	Dependency labels used in my datasets	151
Table 6.12	Details of Training, Development and Testing datasets	152
Table 6.13	The performance of different tools and approaches	153

Table 6.14	The results of multi-lingual parsing using uuparser and graph-based and transition-based algorithms	153
Table 6.15	The fine-grained evaluation results	153
Table 7.1	List of tools and resources have been developed in this study .	168

LIST OF ABBREVIATIONS

Abbreviation	Description
1S	First person – Singular
1PLE	First person - Plural - Epicene
3SN	Third person – Singular – Neuter
3SM	Third person – Singular – Masculine
3SF	Third person – Singular – Feminine
ABL	Ablative
ACC	Accusative
ADJ	Adjective
ADJL	Adjectivalizer
ADV	Adverb
ADVL	Adverbializer
AGR	Agreement
AP	Adjectival Participle
ARG	Argument
ASCII	American Standard Code for Information Interchange
ASS	Associative
AUX	Auxiliary
AVM	Attribute-Value-Matrix
BEN	Benefactive
BILSTM	Bidirectional Long Short-Term Memory
BLEX	Bi-LEXical dependency score
CAR	Cardinal
CAUS	Cause Marker
COMP	Complementiser
COND	Conditional
CONJ	Conjunction
COP	Copula
CRF	Conditional Random Field

DAT	Dative
DEC	Declarative
DIST	Distal
DOM	Differential Object Marking
DUR	Durative
EMPH	Emphatic
EPI	Epicene
F	Feminine
FEATS	Features
FST	Finite-State Transducer
FUT	Future Tense
GEN	Genitive
GEND	Gender
HDT	Hamburg Dependency Treebank
HON	Honorific
HON	Double Honorific
HORT	Hortative
IMP	Imperative
INCL	Inclusive
INESS	Infrastructure for the Exploration of Syntax and Semantics
INF	Infinitive
INS	Instrumental
IRRAT	Irrational
INTJ	Interjection
IOBJ	Indirect Object
LOC	Locative
LAS	Labelled Attachment Score
LFG	Lexical Functional Grammar
LV	Light Verb
M	Masculine Analyzer
MA	Morphological Azad
MLAS	Morphology-aware Labeled Attachment Score
MWTT	Modern Written Tamil Treebank
N	Neuter
NEG	Negative
NER	Named Entity Recognizer
NLP	Natural Language Processing
NLTK	Natural Language ToolKit
NMLZ	Nominaliser

NOM	Nominative
NP	Noun Phrase
NTYPE	Noun Type
NUM	Number
OBJ	Object
OBL	Oblique
ORD	Ordinal
PART	Particle
PASS	Passive
PERM	Permissive
PERS	Person
PL	Plural
POS	Parts of Speech
POSS	Possessive
PP	Postposition Phrase
PRED	Predicate
PRES	Present tense
PROG	Progressive
PRON	Pronoun
PRS	Present Tense
PSP	Postposition
PST	Past Tense
QUOT	Quotative
RAT	Rational
REL	Relativiser
RNN	Recurrent Neural Network
SAN	Sandhi
SEM	Semantic
SER	Singular - Epicene - Rational
SG	Singular
SUBJ	Subject
SVC	Serial Verb Construction
SYM	Symbol
TB	Treebank
TNS-ASP	Tense-Aspect
TTB	Tamil TreeBank
UAS	Unlabelled Attachment Score
UD	The Universal Dependencies
UPOS	Universal Part of Speech

VP	Verbal Phrase
VPART	Adverbial Participle
VTYPE	Verb type
XCOMP	Non-finite clause argument
XLE	Xerox Linguistic Engine
XPOS	Language-specific Part of Speech

TRANSLITERATION SCHEMA

Vowels		Consonants	
அ	a	க்	k
ஆ	ā	ங்	ñ
இ	i	ச்	c
ஈ	ī	ஞ்	ñ
உ	u	ட்	ṭ
ஊ	ū	ண்	ṇ
எ	e	த்	t
ஏ	ē	ந்	n
ஐ	ai	ப்	p
ஓ	o	ம்	m
ஔ	ō	ய்	y
ஔ	au	ர்	r
		ல்	l
		வ்	v
		ழ்	ḷ
		ள்	ḷ
		ற்	ṛ
		ன்	ṅ

Note: Composite characters are formed by adding consonants and vowels together. For instance, Tamil letter க் is transliterated as *ka* as க் = க் (k)+ அ(a) In this way there are 216 composite Tamil letters are formed by composing 18 consonants with 12 vowels, and the composite letters will be transliterated accordingly.