# Computational Approaches to Tamil Linguistics: Scopes and Prospects

**Vasu Renganathan**
University of Pennsylvania
vasur@sas.upenn.edu

## Introduction

This article is an excerpt of my forthcoming book with a relatively identical title to be published by Cre-A, Chennai and it discusses in a nutshell the application of a set of linguistic principles for the processing of Tamil language texts within the four major sub disciplines namely phonology, morphology, syntax and semantics.[1] The two major fields of studies namely 'Natural Language Processing' and 'Computational Linguistics' are understood interchangeably denoting to the same ideas and principles in their own ways of understanding. However, these two disciplines are considered to be distinct in the sense that the former is attributed more toward the language engineering aspects of processing natural language text, where as the latter is attributed more toward the development of the discipline of linguistics with the exclusive aid of computer as a tool. While the former can employ computational principles such as statistical applications, parsing mechanisms and others to effectively process linguistic text for the applications such as machine learning, machine translations and so on, the latter is mostly employed to describe the natural language within the principles of computational algorithms, especially in the fields of lexicography, morphological tagging of texts, identifying the syntactic and semantic structures of the language and so on. Thus, both fields, in essence contribute to each other by many exclusive means in utilizing the electronic

[1] This forthcoming book entitled "Computational Approaches to Tamil Linguistics" deals with my research that I pursued intermittently for the past twenty years concentrating on both Tamil linguistic theories and computational practices and published in a number of international research journals and conference proceedings (cf. Renganathan, 2014, 2010, 2003, 2002, 2001, 1997, 1994, 1993 and 1988). This book, that is expected to be published by the end of August, 2016 might as well be available at the same time along with the proceedings of this conference. The software in its downloadable and executable forms, as illustrated in this book, have their demonstrable working versions at the websites: http://www.tamilnlp.com and http://www.thetamillanguage.com.

systems efficiently to help with the processing and understanding of natural languages by human. In this sense, this work focuses immensely upon defining a comprehensive Natural Language Understanding (NLU) system that can account for the linguistic principles of Tamil language in terms of computational phonology, morphology, syntax and semantics.

**Intricacy of Tamil and accountability by Electronic means**

At the outset, one needs to discuss in detail the limitations of building any natural language system for Tamil due to its inherently complex structure. Tamil, like any other natural languages, contributes immense amount of complications and accounting for all of the intricacies of this language in any substantial level requires many ingenious methods both from linguistic as well as computational points of views. In no context one can, thus, claim all of the complexities of this language can be accounted in any major way. One of the major tasks of dealing with Tamil is accounting for its differences due to its formal literary variety and the informal spoken variety, which have their own form of writing systems. The natural Tamil text that is drawn from different sources including news papers, books and other documents contain within it a combination of both of these varieties and devising a system to understand these differences found to be a daunting task. As far as possible, the systems that are defined in this work account for in a manageable way only the literary variety and the spoken variety is ignored in most cases. One of the reasons for this state of affairs is that the spoken variety can not be accounted for by any particular standard, as in the case of literary variety, due to presence of many dialect forms. This particular behavior of Tamil text does become one of the major limitations of building any comprehensible natural language system for Tamil. The other major issue relating to processing of Tamil texts involves handling words that have multiple shades of meaning with multiple collocational possibilities. Tamil exhibits a very rich historical lexical development from the Sangam period to the modern time. Along the way, internal as well as external linguistic changes due to foreign language influences as well as presence of many genres contributed extensively to the convoluted forms of Tamil words. Thus, building any comprehensible form of natural language understanding system with a large scope would be humanly impossible

as far as Tamil language is concerned. See Renganathan (2011) for a detailed discussion on the historical developents of Tamil from Sangam to modern Tamil.

**Electronic means of studying the sub disciplines of Linguistics**

The sub disciplines of linguistics namely phonology, morphology, lexicography, syntax and semantics are interrelated to each other in a complex fashion and many linguistic theories have attempted to illustrate the relationship between them in a coordinated fashion. One of such work can be attributed to the theory of Lexical Phonology which interrelates the two disciplines namely phonology and morphology in a comprehensive principle called level ordered morphology. This work fully depends on this theory as proposed in the works of cf. Arnoff (1974), Mohanan (1986) and Kaise and Shaw (1985) as well as the syntactic theory of 'Government and Binding' as discussed in Chomsky (1982). Along side, the major computational principles of 'Augmented Network Mechanisms', Tree Adjoining Grammars (cf. Abeille Anne et al 1990) and Prolog's logical methods of processing texts (cf. Clocksin, W.F. and C.S. Mellish 1987) are extensively employed to define an inter-disciplinary system between Tamil linguistics and computational theories. The major applications that are attempted in this work involve development of Text to Speech system, morphological tagger, English to Tamil machine translation system, information retrieval system from Sangam to modern Tamil texts along with a development of English-Verb Tamil dictionary by electronic means. All of these systems which have been developed with a considerable amount of success have demonstrable version at the website: http://www.tamilnlp.com. It is to be stated here that none of these systems are totally independent of each other and they have been developed in such a way that they form a number of related linear processes toward describing the language. The phonological rules, for instance, contribute to the formation of morphems. Morphological structures, in turn, contribute to the syntactic structures. Complex syntactic structures, in consequence, are responsible for understanding the language from the point of view of the principles of semantics. In this respect, any natural language system that is developed along side of the principles of computational algorithms should adhere to these linear processes in one way or another. Along the same line of thought, the proposed work describes within it a number

of software algorithms at every stage and the output of each stage is utilized extensively by the subsequent stages. The basis of any phonological theory is a combination of sounds which contribute in a very subtle way to many linguistic changes in the language from the point of view of morphology, syntax and semantics. Unless this particular behavior is captured by electronic means, understanding and description of linguistic texts in a coordinated fashion becomes an impossible task to accomplish. The combination of the front vowels 'i' and 'e' with its palatal feature, for example, influences the following vowels to occur with the palatal glide 'y'. Similarly, the labial nature of the back vowels 'o' and 'u' require the following vowels to occur with the labial glide 'v'. This type of language specific rules and many more of other complex phonological, morphological and syntactic rules that contribute to the complex structure of words, sentences and consequently the meaning of the language in Tamil need to be accounted for one way or another in all of the electronically definable systems along the line of the linguistic principles. In this sense, this work is an attempt to outline the linguistic rules of the Tamil language at the phonological, morphological, syntactic and semantic levels and consequently map them with that of the computational algorithms using the programming languages Prolog, PHP, Javascript and relevant others.

**Machine understandable form of linguistic strings**

The morphological tagger that is built as part of this system corresponds to the fundamental structure of the overall NLP system that is developed as part of this work. This tagger is capable of recognizing most of the literary Tamil words and output a list consisting of the root form of the words along with suitable tags representing each suffix. Different components of this system have been built primarily based on the ideas proposed in the theory of lexical phonology, which in turn has been primarily developed from the works of Kiparsky (1982), Mohanan (1986), Kaisse and Shaw (1985) and others. The theory of lexical phonology describes the phonological and morphological behavior of natural languages from the points of views different from any previous theories. The goal is to identify the inter-relationship between phonological, morphological and syntactic boundaries of languages in a more convincing and plausible manner than any other competing theories of phonology, which normally concentrate

only on phonological behaviors of natural languages and rules in isolation. In this sense, it would be appropriate to say that this is a theory about both phonology and morphology, rather than just phonology. Keeping in mind these inter-disciplinary ideas, the proposed system produces a machine understandable form of Tamil words in a list form which corresponds to the logical structure as conceivable in the programming language Prolog in terms of 'head' versus 'tail'.

Complex Tamil words such as eṭuttukkoṇṭavarkaḷaiyumā, Is it those who took something?, for instance, is recognized by this system in a list form as [[acc, eTu, pn_refl_hum.pl, inte]] with suitable tags arranged in a systematic fashion. This list structure can be analyzed in Prolog as [[acc| eTu, pn_refl_hum.pl, inte]] with the tag 'acc' being the 'Head' and the rest being the 'Tail'. This can further be analyzed with the same principle by segmenting the string of tags being Head and Tail. Thus, this list structure that this system depends on in most of the components forms the basis of machine understandable string of Tamil texts. It is demonstrated how this fundamental structure constitutes a convenient way of processing Tamil texts for other applications such as information retrieval system, machine translation, development of morphological tagger and subsequently the development of text to speech systems. Some of the major advantages of conceiving the strings of Tamil texts in this type of list structure is that it not only eliminates the undesirable low level linguistic information such as sandhi rules, but also closely aligns to the logical method of analysis of natural languages by computer algorithms. When all of the texts are converted into this type of list structures, both word internal, intra-sentences across the words as well as inter-sentential information can very well be accounted for by the same fundamental principle of 'Head' versus 'Tail'. Thus, construction of parsing systems can also be accomplished by this kind of computer understandable form of Tamil texts. The second stage of this system is a principle-based parser, constructed solely based on the notions of the theory of Government and Binding (GB) (Chomsky, 1982), which examines whether all of the elements within the output list produced by the CF parser conforms to the basic principles as outlined in this theory. One of the important features of the GB theory is its universality, as the principles that constitute it are common to all human languages and

any difference among languages are accounted for by setting suitable parameters. (cf. Ghemri, 1991: 111).

Any typical input and output from morphological and syntactic components of this system can be construed as below:

**Input in Tamil string:**

aracan mandtiriyiTam ivarkaLukku evvaLavu paNam koTukkalaam enRu keeTTaan

**Translation:**

"The king asked the minister how much money he should give to the poets"

**Output in a List form from the Morphological Tagger:**

[["nom","aracan","noun"],["hloc","mandtiri","noun"],
["dat","ivarkaL","noun"],["mass","evvaLavu","ques"],
["nom","paNam","noun"],["possi","koTu"],["comp","enRu"],
["pa","keeL","3sgmas"]]

**Output from Context-Free Grammar parser:**

[[[["nom","aracan","noun"]]],[[["hloc","mandtiri","noun"]]],
[[["dat","ivarkaL","noun"]]],[[["mass","evvaLavu","ques"],
["nom","paNam","noun"]]],[[["possi","koTu"]]],
[[["comp","enRu"]]],[[["pa","keeL","3sgmas"]]]]

As already stated, interpretation of Tamil texts in the form of this type of list structures become the basis of other applications which rely on all of the subtle information related to morphology, syntax and semantics within any sentential as well as inter-sentential forms. The system defined in this work accounts for almost all of the morphological rules of Tamil in Prolog and is capable of tagging Tamil text to the maximum extent possible, provided a comprehensive lexicon is made available as part of the system. This system has been tested with a comprehensive word list from the Madras University Lexicon as well as a list of manually encoded Tamil words. The sample input and output

of this tagger is made available in the website with text choosing from a number of Tamil novels and Tamil news papers.

The important task, however, is to arrive at a standard for defining tags for every conceivable morphemes of the language and subsequently to arrange them in a linier fashion so suitable code can be devised to navigate through these tags in a logical fashion. In this sense, what is attempted in this work is to write a grammar of Tamil conforming to the principles of both linguistic as well as the computational theories. The major task of this type of systems is accounting for the humanly inconceivable complex structures as well as many ambiguous interpretations. Further, accounting for all of the complex set of words and their roles in defining meaning in various collocations is yet another daunting task that this system faces as none of the lexicon so far built provides a very comprehensive information accounting for all of the complex nature of collocational possibilities. In this sense, it is to be stated that any successful full-fledged natural language processing system is possible only when the linguistically complex systems such as lexical, syntactic and semantic systems are definable in terms of machine readable form in one way or another. Any natural language text, in this context, contains within it more convoluted and multifaceted information that anyone can imagine within any set of accountable logical rules.

**Prolog's List Manipulation and Set Theoretical Notations**

List structures conform to Prolog's built-in logical methods of Head and Tail. This method is assumed to be one of the easiest and fastest methods in the computational processing of natural language sentences, and it has been studied widely by many researchers. Studies that discuss the concept of lists for processing natural languages can be found in a number of works including Head-driven Phrase Structure Grammar (HPSG) (Sag and Pollard, 1986), PATR-II (Shieber, 1986), and Unification-based grammars (Shieber, 1986). Also, the programming languages LISP and Prolog are powered with several list-handling capabilities, enabling one to construct NLP systems in a logical manner conforming to the elements in ordered lists.

An implemented system that can process Tamil sentences using a set of user-defined and built-in list handling predicates written in the programming language Prolog is described in this work with suitable examples. This system makes use of the predicates from morphological and syntactic components. It is constructed in such a manner that it can take any Tamil sentence as input, convert it into its corresponding list form and finally store the list in a separate database. When a query based on the facts as stored in the input sentences is entered, the system converts the query also into its corresponding list form and compares the elements inside the list to evaluate the facts. Further, the output Tamil sentences are generated using a separate module that is designed to make words and sentences by making use of the sentence-generation algorithm. Thus, the understanding of natural language by this system is construed as having to compare constituent structures that are stored in the form of ordered lists of words and suffixes as opposed to concatenated string words and affixes.

**List Manipulation technique and understanding Tamil Words**

Recall that the morphological component of this system recognizes every input word and stores information about suffixes and root forms in a list form. To cite an example, a word such as keeTTukkoNTirukkiRaaraa? 'has he been listening?' is converted by the morphological component of this system into a list consisting of verb and the corresponding suffixes with suitable tags as in:

[pr_prog, keeL, 3mas.sg, inte]

The term word is defined in Prolog in such a way that it contains all of the information in the form of mark-up tags along with the root form. The members of this list are related to each other based on the unique relationship of Head versus Tail. For example, in the notation [H||T], when the variable H is instantiated with the word pr_prog the variable T is instantiated with rest of the elements namely keeL, 3mas.sg and inte. Subsequently, one can also identify the main word [A,B||T] with the variables A and B being instantiated with the tag pr_prog and the verb keeL 'listen' respectively. Then, the variable T is instantiated with the rest of the tags. Thus, with this basic idea of Head versus Tail, it is possible to construe input words as well as sentences based on a

number of different criteria, such as (a) type of words, (b) collocational words, (c) presence or absence any suffix and so on. This type of task is carried out using a number of user-defined list manipulation Prolog predicates like head(L,A), subset(A,L), sublist(L1,L2), and so on. The convenience of this method of analyzing natural language sentences is that almost all of the natural language concepts – syntactic relationships such as subject, object, predicate, argument, adjunct etc., as well as the semantic relationships such as synonymy, hyponymy, homonymy – can be accounted for. Thus, the list processing technique is a parallel technique that we employ to construct a program that can understand natural languages similar to how humans understand language.

The sentence-generation part of this system namely 'adgener.pro', in turn, takes this list as input and produces the corresponding output word in a string form. Thus, with the use of these list structures, it is possible for one to generate any desired output simply by replacing the morphological tags with other related ones to produce a desired set of output sentences. For example, in order to generate a completive form of the word keeL 'listen', with the human plural suffix, all one needs to perform is to replace the respective tags as illustrated below.

[pr_compl, keeL, 3mas.pl, inte]

This list can now be used by the generation part of the system to output the corresponding Tamil word keeTTuviTukiRaarkaLaa 'are they listening to it?'. Thus, the two intriguing aspects of natural language such as construction of morphologically and grammatically well-formed structures, as well as understanding them based on their interrelationships are accounted for by this system by conveniently transforming the natural language input sentences into machine understandable list structures.

**Syntactic Components and List Manipulation Techniques**

The syntactic component of this system consists of a Context Free (CF) grammar constructed by adhering to the ideas as outlined in Prolog's Definite-Clause Grammar

(DCG). This is a bottom-up parser, which parses input sentences from left to right. All the words and suffixes required are converted into corresponding tags and stored in a sequential list form. This list structure parallels every natural language input sentence and is the crucial part of the NLP system, which infers meaning in exactly the same manner as any human might understand a natural sentence. For instance, humans identify words and phrases by their endings. The phrase 'avnuTaiya tantaikku' 'for his father' is understood as a dative noun phrase only because the head noun ends with a dative suffix. Similarly, this system infers information from the list [[gen,avan],[dat,tantai]] using the tag 'dat', which occurs as the head of the list that follows the genitive phrase.

**Defining Word, Phrase and Sentences in the context of List structure**

The term 'word', thus, is defined in Prolog as a list consisting of suitable tags for suffixes and the root form of the input word enclosed within two square brackets. The term 'phrase', on the other hand, consists of more than one word enclosed within square brackets. Thus, the phrase ennuTaiya tampi 'my younger brother' is tagged as follows.

```
[       [gen,ndaan],[nom,tampi]  ]
Phrase  Word        Word         Phrase
```

Correspondingly, a sentence is assumed to consist of a group of noun and verb phrases that are marked with notations such as npm() and vpm() representing each NP-MAX and VP-MAX respectively. Thus, the parsed structure of the sentence ennuṭaiya tampi veekamaakap paṭippaan 'My younger brother reads fast' is generated as below.

```
[ [   [   [gen,ndaan],[nom,tampi] ]   ],
S NPM Phr Word        Word         Phr NPM

[   [   [adv,veekam],[fut,paTi,3sg.mas] ]   ]     ]
VPM Phr Word                          Phr VPM   S
```

**Registers, Unification and construction of Machine Translation systems**

With a list structure being the fundamental form for the equivalent linguistic text, the subsequent task to comprehend the structure is devising a system to parse the list structures the way humans understand a sentence. Any system with the rules of the Recursive Transition Network (RTN) formalism accepts all possible syntactically acceptable structures as licensed by any given CFG grammar. The advantage of ATN, according to Winograd (1972:43) is that its operation appears to be closer to the actual humans' use in understanding a language. The structure of ATN rules as shown in Obermeier (1989: 76) is reproduced below.

'ATNs are similar to recursive transition networks but have three additional features: registers, which can store conditional information on a global basis, regardless of which particular sub-network is being processed; conditions, which let arcs be selected if registers indicate certain conditions; and actions, which let arcs modify the structure of data.'

The three components registers, conditions and actions of ATN are constructed in the form of Prolog's rule formalisms in order to account for a restricted domain of relative clause constructions in English and consequently translate them to Tamil. Tamil does not have reduced relatives, and also the equivalent for all English Wh words in Tamil is a suffix, –a. This requires a creation of two different rules for English and Tamil separately. For instance, consider the following example translations from the Machine Translation system,

1. She tells me a story which almost does freeze my blood
The system translates this sentence into Tamil as below.

1a. avaḷ ennuṭaiya rattatt-ai eer̲akkur̲aiya uraiyavai-kkir̲-a
she my blood-acc almost freeze - pr. Wh
kataiy-ai ena-kkuc col-kir̲-āḷ
story-acc me-to say pr. she(personal termination)

Sentence 1 is an appositive relative clause that does not have a direct equivalent in Tamil, whereas 1a is the equivalent structure that this system translates to. Now consider the following ungrammatical sentences that any pure RTN grammar (as opposed to ATN grammars, which implement suitable registers) would translate in the same way as it does for the grammatical sentences as shown above.

1b. *She tells me a story who/where/when/what almost freeze my blood.

This is because all the Wh words in English have the suffix -a as their equivalent in Tamil and there is no concord relationship in terms of gender between the suffix -a and the head noun. In order to make the system identify the ungrammaticality of sentences like 1b, an user-defined prolog predicate relative(Head,Wh) is used. This register is constructed in such a way that it can unify the gender of the head noun with that of the gender of Wh pronouns. Similar technique is used to navigate between the structures of English and Tamil to produce a comprehensive machine translation system. Identical to this construction, following relative clause sentence is also dealt with in this system adhering to the principles of registers and unification.

2. a muscular dome shaped diaphragm, present in mammals, separates the thoracic from the abdominal cavity.

2a. neñcaraiyai vayirraraiyiliruntu pirikkinra pālūṭṭikaḷil irukkinra oru tacaiyiṉāl āṉa kūmpu vaṭivuḷḷa utara vitāṉam.

Following is yet another sample set of input sentences that this system is capable of translating.[2]

Rabindranath's poem Gardener

if you would have it so, i will end my singing

if it sets your heart aflutter, i will take away my eyes from your face

[2] This English-Tamil translation is system and the morphological tagger are testable at the URL: http://www.thetamillanguage.com/tamilnlp/

if it suddenly startles you in your walk, i will step aside and take another path

if it confuses you in your flower_weaving, i will shun your lonely garden

if it makes the water wanton and wild, i will not row my boat by your bank

Translated Tamil text in Roman script

tooTTakkaarar

ndiingkaL itai appaTi vaittiruppiirkaL enRaal ndaan ennuTaiya paaTuvatai ndiRuttuveen

itu ungkaLuTaiya itayattai paTapaTakka ceykiRatu enRaal ndaan ennuTaiya kaNkaLai ungkaLuTaiya mukattilirundtu eTuttuviTuveen

itu ungkaLai ungkaLuTaiya ndaTaiyil tiTiirenRu taTumaaRacceykiRatu enRaal ndaan ndakarndtupooy veeRoru vazhiyai eTuttuviTukiReen

itu ungkaLai ungkaLuTaiya puuppaRippatil kuzhappukiRatu enRaal ndaan ungkaLuTaiya taniyaana tooTTattai tavirttuviTuveen

itu taNNiirai kuzhappam maRRum kalakkamaaka ceykiRatu enRaal ndaan ennuTaiya paTakai ungkaLuTaiya kaRaikku pakkattil tuTuppuppooTamaaTTeen

**Concluding remarks**

This work is a detailed exposition of the issues relating to construction of a comprehensive Natural Language Understanding system that can be made use of by many of the natural language tasks. Prolog's set theoretical principles along with the principles of the theories of Lexical Phonology and Government and Binding are defined in an inter-disciplinary manner in order to bring the two relatively distinct fields, namely Linguistics and Computer Science, come together. The development of prototype applications of an English–Tamil machine translation system, Man-Machine Tamil interface application, information retrieval system, morphological tagger are illustrated in this work with suitable examples from corresponding computer algorithms written in the programming language Prolog. Even though this study has a relatively limited amount of scope, the countless lines of programing code that are discussed in this work can be a good starting point in many future applications that intend to be built with any identical scope. The code and working examples of these applications are made available at the website http://www.thetamillanguage.com/tamilnlp and http://www.tamilnlp.com. The

application of tagged corpora that this system can be used to build may not be limited to use within the fields of language and literature, but that it can be extended to every conceivable area of linguistic and literary studies. Analysis of text styles of any historical period, obtaining statistical information of different kinds, pedagogical applications, dictionary and thesaurus making, tracing the trajectories of linguistics changes etc., are some of the promising areas which would require the use of a good corpus and the morphological tagged system. In this respect, the morphological system and the set theoretical method of analysing the linguistic texts can be employed as a fundamental resource for any future endeavors relating to processing of Tamil by electronic means.

**References**


Abeille Anne, Katheleen Bishop, Sharon Cote and Yves Schabes (1990). A Lexicalized Tree Adjoining Grammar for English. MS-CIS-90-24, Department of Computer and Information Science, University of Pennsylvania.

Aronoff, Mark (1974). Word-Structure. Unpublished Ph.D. dissertation, MIT, Massachusetts.

Chomsky, Noam (1970). Remarks on Nominalization. In Roderick A. Jacobs and Peter S. Rosenbaum (eds.) *Readings in English Transformational Grammar.* Ginn and company, Massachusetts.

Chomsky, Noam. (1982). *Lectures on Government and Binding*, Foris Publications.

Clocksin, W.F. and C.S. Mellish (1987). *Programming in Prolog. (3rd ed.)* Springer-Verlag: New York.

Collins, Allan M. and Quillian, M.R. (1969). 'Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8, 240-247.

Cruse, D.A., (1986). *Lexical Semantics*. Cambridge: Cambridge University Press.

Daelmans, Walter. et al. (1992). "Inheritance in Natural Language Processing". *Computational*

Gazdar, Gerald. and Chris Mellish. (1989). *Natural Language Processing in Prolog.* Addison-Wesley Publishing Company: Wokingham.

Ghemri, Lila. (1991). “Specification and Implementation of a GB parser". In C. Brown and G. Koch (eds.) *Natural Language Understanding and Logic Programming III*. Elsevier Science Publishers B.V. (North-Holland).

Kaisse, E.M. and P.A. Shaw. (1985). “On the theory of lexical phonology". *Phonology year book* 2:1-30.

Karunakaran, K. and M. Jayakumar (1988) (eds.). *Translation as Synthesis: A search for a new gestalt*. Bahri Publications: New Delhi.

Kiparsky, Paul (1982). “From Cyclic Phonology to Lexical Phonology." In H. Van der Hulst and N.Smith (eds.) *The Structure of Phonological representations 1.* Dordrecht: Foris.

Mohanan, K.P. (1986). *The Theory of Lexical Phonology.* Dordrecht: Reidel.

Obermeier, Klaus K. (1989). *Natural Language Processing Technologies in Artificial Intelligence*, Ellis Horwood Limited, England.

Renganathan, Vasu (2014). "Computational Phonology and the Development of Text-to-Speech Application for Tamil.", Tamil Internet Conference, 2014, Pondicherry University: Pondicherry. (http://www.infitt.org/ti2014/papers/135_VasuR_final.pdf).

_____. (2011). *The Language of Tirumular’s Tirumantiram: A medieval Saiva Tamil religions Text.* Ph.D. Dissertation, University of Pennsylvania.

_____. (2001). "Development of Part-of-Speech Tagger for Tamil", Tamil Internet Conference, 2001 (http://infitt.org/drupal7a/TIconferencepapers/TI2001/vasur.pdf).

_____. (2002). "An interactive approach to development of English to Tamil Translation machine Translation system on the web", Tamil Internet Conference, 2002. (http://infitt.org/drupal7a/TIconferencepapers/TI2002/15VASUR.PDF).

_____. (1993). "Tamil Anaphor Binding in Distant Clauses". In Davison, Alice and Frederick M. Smith. South Asian Studies Program: University of Iowa, Iowa city.

_____. (1993). “A logical approach to development of natural language understanding system for Tamil." *PJDS* 3:1,53-64).

_____. (1994). “On the nature of Wh-trace in Tamil Relative Clauses." *IJDL* Vol. XXIII No. 1:44-64 January 1994.

_____. (1997a). “A Lexical Phonology Approach to Processing Tamil Words by Computer." *IJDL* Vol. XXVI No. 1, January 1997.

_____. (1997b). “Significance of Creation and Use of Corpus of Modern Tamil Prose Text through the Web." Paper presented in the Interntional Symposium for Tamil Information Processing and Resources on the Internet, National University of Singapore, Singapore, May 1997.

_____. (1988). “Human Aided Machine Translation (Problems and Perspectives)." in Karunakaran and Jayakumar (eds.).

Shieber, S. (1986). “An Introduction to Unification-based approaches to grammar". CSLI Lecture Notes 4: Chicago University. Press.

Winograd, T. (1983). *Language as a Cognitive Process.* Addison-Wesley.