MORPHOLOGICAL ANALYZER FOR CLASSICAL TAMIL TEXT: A RULE-BASED APPROACH

R. Akilan[1] and E. R.Naganathan[2]
[1]Research and Development Centre, Bharathiar University, Coimbatore, India
[2]Department of Computer Science and Engineering, Hindustan University, Chennai, India
E-Mail: akilan.rp@gmail.com

ABSTRACT

Morphological Analyzer is the essential and basic tool for building any language processing application. Morphological Analysis is the process of providing grammatical information of a word given its suffix. Morphological Analyzer is a computer program which takes a word as input and produces its grammatical structure as output. A Morphological analyzer will return its root/stem word along with its grammatical information depending upon its word category. Classical Tamil Morphology is very rich and agglutinative language. Morphological analyzer is the tool needed for the following Natural Language Processing applications like information retrieval, search engine, spell checkers, grammar checker, machine translation, dictionary making systems, information extraction and retrieval, content analysis and question answering systems. The rule-based approach has successfully been used in developing many natural language processing systems. The present paper deals with the design and development of morphological analyzer for Classical Tamil and shows its results at end.

Keywords: morphological analyzer, classical Tamil, NLP, Tamil.

1. INTRODUCTION

Morphology is the study of internal structures of a word. Morphological analysis is the process of segmenting words into morphemes and analyzing the word formation. It is a primary step for various types of text analysis of any language. Morphological analyzers are used in search engines for retrieving the documents from the keyword. The morphological analyzer increases the recall of search engines. It is also used in speech synthesizer, speech recognizer, lemmatization, noun decompounding, spell and grammar checker and machine translation. Classical Tamil language is morphologically rich languages needs deep analysis at the word level to capture the meaning of the word from its morphemes to generate word. In general Tamil language is postpositional inflected the root word. Each root word can take a few thousand inflected word forms. Classical Tamil language takes both lexical and inflectional morphology. Lexical morphology changes the word meaning and its class by adding the derivational and compounding morphemes to the root. Inflectional morphology changes the form of the word and adds additional information to the word by adding the inflectional morphemes to the root [1].

Morphological Analyzers is available for Modern Tamil Language such as IL-ILMT, AU-KBC, CIIL, Mysore, etc. There are various approaches like corpus based approach, stochastic models and hybrid approaches. In this context, Classical Tamil further presents a challenge in developing Morphological Analyzer as the language is highly inflectional and morphologically rich.

2. RELATED WORKS IN CLASSICAL LANGUAGES

The Morphological analyzer has been developed for the Classical languages for various methodologies and approaches. Arabic Morphological Analysis and Generation has developed by Kenneth R. Beesley at the Xerox Research centre Europe it was built using Xerox Finite-State Technology. Hebrew Morphological Analyzer developed by Shlomo Yona using Finite-state automata. They developed a Morphological Analyzer for un dotted Hebrew words that is based on Finite-state linguistically motivated rules and a broad coverage lexicon [2]. Greek Morphological Analyzer was developed by David W. Packrd under the Innovative Projects in University Instruction, University of California. The goal was to develop a new textbook and curriculum for teaching ancient Greek to American students [3]. Latin parser and translator were developed by Adam McLean to translate form Latin to English [4].

Sanskrit Morphological Analyzer was developed by the Special Centre for Sanskrit Studies, Jawaharlal Nehru University, New Delhi that identifies and analyzes inflected noun forms and verb-forms in any given sandhi-free text [5]. Chinese Morphological Analyzer was developed by Tseng and Chen developed a Morphological Analyzer for Chinese; their task is to automatically analyze the morphological structures of compounds words. The morphological structures of compound words contain essential information regarding their syntactic and semantic characteristics [6]. Persian Morphological Analysis A finite-state Morphological Analysis of Persian is developed by Karine Megerdoomian, Department of Linguistics, and University of California, USA. The analyzer describes a two-level Morphological Analyzer for Persian using a system based on the Xerox Finite State tools. [7].

3. EXISTING MORPHOLOGICAL SYSTEMS FOR TAMIL

Some of the attempts have been made for Morphological Analyzer for Tamil. It is listed as below

3.1 Morphological Analyzer for Tamil by Prof. S. Rajendran: This was one of the very first efforts towards building a morphological analyzer for Tamil. It was initiated by anusaraka group. Tamil University prepared this morphological analyzer for Tamil to translate Tamil into Hindi at the word level.

3.2 Anna University K.B. Chandraseakr Search Centre (AUKBC) has developed the Morphological Parser for Tamil. The API Processor of AUKBC makes use of the finite state machinery like PC Kimmo. It parses, but does not generate.

3.3 Morphological Generator and Analyzer for Tamil by Ms. Vaishanvi have built generators and analyzers for Tamil morphology. The generator implements the item and process model of linguistic description. It works by the synthesis method of PC Kimmo. The analyzer uses a hybrid model for Tamil. It is theoretically rooted in a blend of IA and IP models of morphology. It constitutes an in-built lexicon and involves a decomposition of words in terms of morphemes within the model to realize surface well-formed words-forms.

3.4 Resource Centre for Indian Language Technological Solutions (RCILTS), Anna University Morphological analyzer for Tamil: [17] Resource Centre for Indian Language Technological Solutions-Tamil, Anna University, Chennai has prepared a morphological analyzer for Tamil named as Atcharam. It takes a derived word as an input and separates it into root word and associated morphemes. It uses a dictionary of 20,000 root words based on fifteen categories.

3.5 Prof. M. Ganesan has developed a Morphological Analyzer for Tamil. The Analyzer uses phonological and morphophonemic rules and takes into account morphotactic constraints of Tamil. An efficient morphological parser has also been built.

3.6 Prof N. Deivasundarams has developed a Morphological Parser for Tamil. The parser built for the MenTamil Tamil Word Processor. He makes use of phonological and morphophonemic rules and morphotactics constraints of the language for developing his parser [15].

4. CHALLENGES IN MORPHOLOGICAL ANALYZER FOR CLASSICAL TAMIL

Tamil is a classical language which belongs to the Dravidian language family. Tamil literature has existed for over two-thousand years. The morphological structure of Classical Tamil is quite complex since it inflects to person, gender, and number markings and also combines with auxiliaries that indicate aspect, mood, causation, attitude etc in verbs. A single verb root can inflect for more than two-thousand word forms including auxiliaries. Noun root inflects with plural, oblique, case, postpositions and clitics. A single noun root can inflect for more than five hundred word forms including postpositions. The root and morphemes have to be identified and tagged for further language processing at word level. The structure of verbal complex is unique and capturing this complexity in a machine analyzable and generatable format is a challenging job [11]. The formation of the verbal complex involves arrangement of the verbal units and the interpretation of their combinatory meaning. Phonology also plays its part in the formation of verbal complex in terms of morphophonemic or *sandhi* rules which account for the shape changes due to inflection [20].

5. RULE BASED APPROACHES IN NLP

The rule-based approach has successfully been used in developing many natural language processing Applications [19]. The linguistic knowledge acquired for one natural language processing system may be reused to build knowledge required for a similar task in another system. Systems that use rule-based transformations are based on a core of solid linguistic knowledge. The advantage of the rule-based approach over the corpus-based approach is clear for: less-resourced languages, for which large corpora, possibly parallel or bilingual, with representative structures and entities are neither available nor easily affordable, and for morphologically rich languages, which even with the availability of corpora suffer from data sparseness. These have motivated many researchers follow the rule-based approach in developing natural language processing Analysis and Applications.

6. METHODOLOGY

6.1 Data collection and dictionary

The dictionary plays important role in developing Morphological Analyzer to identify the roots words and its grammatical categories [14]. The primary data for Morphological Analyzer have collected from classical Tamil texts. From this, the root word list has prepared and verified by language experts. The root word list is in XML file format. The root words are collected from the authentic editions of Classical Tamil texts. Collected the words from the authentic editions and using Language Analysis tools pre-processing the words and get the root words for Classical Tamil texts.

6.2 Tagsets

Parts of speech tagging assigns grammatical category of the language. A POS tagset is developed on the basis of the information from a particular language [18]. The tagset consist finite tags and the information extraction should be infinite. The classical Tamil tagset has developed which includes the following categories: Case Marker(CM) (Nominative (CMN), Accusative case (CMA), Instrumental case (CMI), Associative case and sociative case (CMS), Dative case (CMD), Genitive Case (CMG), Locative Case (GML), Clitics (CL), Conjuction (CJ), Demonstrative (DEM), Noun (NN), Particle (PAR), Postposition (POS), Pronoun (PR), Verb (VB), Tense Marker (Present Tense (PTM), Past Tense (PTM) and Future Tense (FTM).

6.3 Rules

The rule based approach is used to derive a given grammatical form is here called a "formation mechanism". The grammatical concepts discussed in this study utilize one or more of the following give formation mechanisms: use of the base stem, stem mutation, suffixation, extension and periphrasis. Nominal and verbal roots serve as bases for adding different types of affixes, and the affixes thus added are commonly known as "suffixes"[8]. Those suffixes here are simply referred to as "markers" because they mark grammatical concepts.

6.4 Rules for verbs

Verb form takes tense marker, person, number, gender markers (PNG). There are no multiple meaning features of tense markers and PNG markers. But it provides many conjugated forms form the verbs. It is well known that in almost all natural languages, verbs are considered to be the most important part of speech. Verbs play an important role in any languages. As Tamil verbs are inflected to various grammatical categories the bulk of Tamil parts of speech dealt with verbs are necessary

1. Check the root word dictionary { if 'yes' assign the appropriate tag}
2. Check the suffix word { an̲, ān̲, ar, atu, an̲a, pa, mār, pa,ṭu, ā, ku, ṭu, tu, r̲u, en̲, ēn̲, al, āḷ, am, ām, em, ēm, kum, ṭum, tūm, r̲um, i, ai, āy, ir, īr}
3. Remove the suffix
4. Check the next suffix { if the suffix 't, ṭ, r̲, in̲, tt, nt, n̲' } split and assign tag 'PATM'}
5. Else if the suffix 'kir̲, kkir̲' split and assign tag ' PRTM'
6. Else if the suffix 'v,p,pp' split and assign tag 'FTM';
7. If the suffix {PATM, PRTM, FTM} Assign the previous tag as PNG
8. Check the remaining word in the root word dictionary
9. if yes { assign the tag as 'VB' }
10. else if {Check the remaining suffix { if the suffix 'ṇ' }}
11. Replace the suffix 'ḷ' instead of 'ṇ'
12. Tag the word as 'VB'
13. Else if Check the remaining suffix { if the suffix 'n̲'}
14. Replace the suffix 'r̲' instead of 'n̲'
15. Tag the word as 'VB'
16. Else if {Check the remaining suffix { if the suffix is 'consonants'}}
17. Add the end of the suffix 'u'
18. If no { display the remaining text}
19. Tag the word as 'VB'
20. Else if {Check the remaining suffix { if the suffix 'ṭ'}
21. Replace the suffix 'ḷ' instead of 'ṭ'
22. Tag the word as 'VB'
23. Else if { Check the remaining suffix { if the suffix 'r̲' }}
24. Replace the suffix 'l' instead of 'r̲'
25. Tag the word as 'VB'
26. Else if {Check the remaining suffix { if the suffix 'a' }}
27. Replace the suffix 'ā' instead of 'a'
28. Tag the word as 'VB'
29. Else if {Check the remaining suffix { if the suffix 'n̲' }}
30. Replace the suffix 'l' instead of 'n̲'
31. Tag the word as 'VB'

32. Stop

Ex. உண்டான் – உண்/VB + ட்/PATM + ஆன்/PNG

ṇṭāṉ - uṇ + ṭ + āṉ - uṇ/VB + ṭ/PATM + āṉ/PNG

ஈங்கேதலைப்படுவன் உண்டான் தலைப் பெயின் (கலி. 64:24)

īṅkētalaippaṭuvaṉ uṇṭāṉ talaip peyiṉ (kali. 64:24)

6.5 Flowchart

The Figure-1 illustrates the Morphological Analyzer flowchart. The tool accepts the Tamil text corpus as an input, which is converted into transliteration script. Then the words are tokenized. After tokenization, at the first stage the algorithm will check the words in root dictionary; if it is available it assigns the appropriate grammatical category. If it is not available, it goes for rules. As per the rules of classical Tamil, words are analyzed. The analyzed words are separated into suffixes and root words. Suffixes are tagged as per the rule. Root words again go for rules. The above mentioned procedure is repeated till the root word is found in the dictionary. If root word is not available in the dictionary, the analyzer assigns it as an unknown category. The result is converted and displayed in Unicode Tamil.

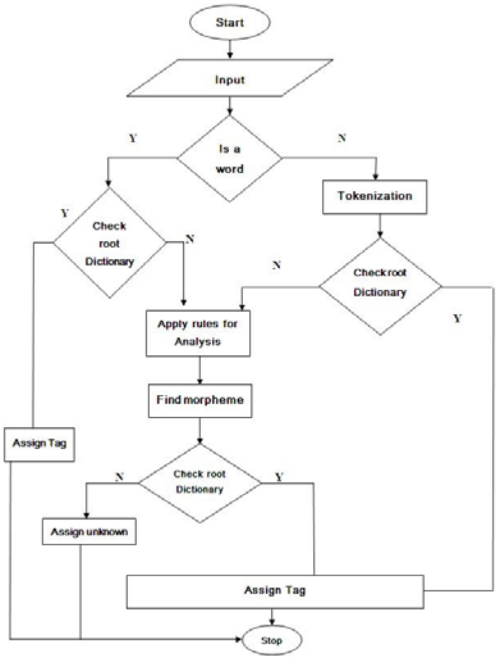
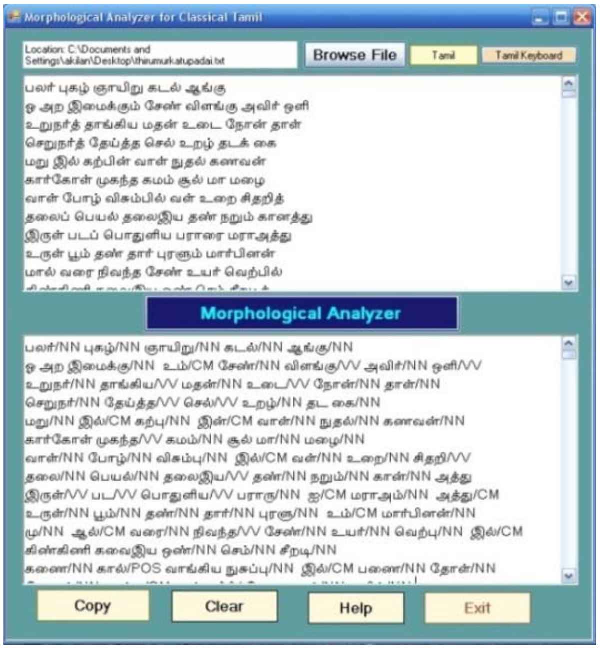


Figure-1. Procedure for morphological analyzer.

The following Figure-2 shows the graphical user interface of the Morphological Analyzer.



Location: C:\Documents and Settings\akilan\Desktop\thirumurkatupadai.txt

பலர் புகழ் ஞாயிறு கடல் ஆங்கு
ஓ அற இமைக்கும் சேண் விளங்கு அவிர் ஒளி
உறுநர்த் தாங்கிய மதன் உடை நோன் தாள்
செறுநர்த் தேய்த்த செல் உறழ் தடக் கை
மறு இல் கற்பின் வாள் நுதல் கணவன்
கார்கோள் முகந்த கமம் சூல் மா மழை
வாள் போழ் விசும்பில் வள் உறை சிதறித்
தலைப் பெயல் தலைஇய தண் நறும் கானத்து
இருள் படப் பொதுளிய பராரை மராஅத்து
உருள் பூம் தண் தார் புரளும் மார்பினன்
மால் வரை நிவந்த சேண் உயர் வெற்பில்

பலர்/NN புகழ்/NN ஞாயிறு/NN கடல்/NN ஆங்கு/NN
ஓ அற இமைக்கு/NN உம்/CM சேண்/NN விளங்கு/VV அவிர்/NN ஒளி/VV
உறுநர்/NN தாங்கிய/VV மதன்/NN உடை/VV நோன்/NN தாள்/NN
செறுநர்/NN தேய்த்த/VV செல்/VV உறழ்/NN தட கை/NN
மறு/NN இல்/CM கற்பு/NN இன்/CM வாள்/NN நுதல்/NN கணவன்/NN
கார்கோள் முகந்த/VV கமம்/NN சூல் மா/NN மழை/NN
வாள்/NN போழ்/NN விசும்பு/NN இல்/CM வள்/NN உறை/NN சிதறி/VV
தலை/NN பெயல்/NN தலைஇய/VV தண்/NN நறும்/NN கான்/NN அத்து
இருள்/VV பட/VV பொதுளிய/VV பராரு/NN ஐ/CM மராஅம்/NN அத்து/CM
உருள்/NN பூம்/NN தண்/NN தார்/NN புரளு/NN உம்/CM மார்பினன்/NN
மு/NN ஆல்/CM வரை/NN நிவந்த/VV சேண்/NN உயர்/NN வெற்பு/NN இல்/CM
கிண்கிணி கவைஇய ஒண்/NN செம்/NN சீறடி/NN
கணை/NN கால்/POS வாங்கிய நுசுப்பு/NN இல்/CM பணை/NN தோள்/NN

Figure-2. Graphical User Interface for Morphological Analyzer.

7. ISSUES IN CLASSICAL TAMIL TEXTS

The word form is in different function. In some cases the word form are same and syntactic structure are same. The grammatical category is depended on syntactic structure of the word. In some cases the word form and syntactic structure both are same this type of post positions highly completed. The machine cannot analyze the word. In this case we need a complete sentence analysis. For examples illustrate the issues of raised in the following Classical Tamil literatures

Ex.

a) கோல் கொண்டு அலைப்பப் படீஇயர் மாதோ (நற்.58)

kōl koṇṭu alaippap paṭīiyar mātō (naṟ.58)

b) வெண் கோடு கொண்டு வியல் அறை வைப்பவும் (நற்.114)

veṇ kōṭu koṇṭu viyal aṟai vaippavum (naṟ.114)

c) புலம்பு கொண்டு உறையும் புன்கண் வாழ்க்கை (நற்.124)

pulampu koṇṭu uṟaiyum puṉkaṇ vāḻkkai (naṟ.124)

In the song (naṟ.58) 'kōṭu' means with here it is post position (PP) and the song (naṟ.114) 'kōṭu' means take here it is verbal participle (VP), in such cases machine cannot analyzed the words, in this cases we need a complete sentence analysis.

8. RESULT AND DISCUSSIONS

The Morphological Analyzer for Classical Tamil text is developed using rule-based approach, this paper takes the successful efforts and first attempts for Classical Tamil texts. The procedure was implemented the major grammatical categories in Classical Tamil. For the testing and evaluation purpose 3257 words have been taken as input of the morphological analyzer, it produces the result of 2706 (83 %) words are analyzed correctly 359 (11%) of words are analyzed wrongly and 194 (6%) of words are un analyzed.

The Table-1 shows that some examples of words and their assigned category.

Table-1. Words and their categories

S. No.	Before analysis	After analysis
1	வண்டும் vaṇṭum	வண்டு\NN உம்\CM vaṇṭu\NN um\CM
2	மூன்றலங்கடையே mūṉṟalaṅkaṭaiyē	மூன்று அலம் கடை ஏ\CL mūṉṟu alam kaṭai ē\CL
3	திறத்தான் tiṟattāṉ	திற\VB த்த்\PTM ஆன்\PNG tiṟa\VB tt\PTM āṉ\PNG
4	மருங்கினால் maruṅkiṉāl	மறுங்கு\VB இன்\ ஆல் maṟuṅku\VB iṉ\ āl
5	கிளைகல் kiḷaikal	கிளை\VB கல்\PL kiḷai\VB kal\PL
6	கரும்பிற்கு karumpiṟku	கரும்பு\NN இன்\CM கு\CL karumpu\NN iṉ\CM ku\CL
7	செய்தான் ceytāṉ	செய்\NV த்\PATM ஆன்\PNG cey\NV t\PATM āṉ\PNG
8	அகத்தை akattai	அகம்\NN அத்து\OM ஐ\CM akam\NN attu\OM ai\CM
9	இன்மையொடு iṉmaiyoṭu	இன்மை\VB ஒடு\CL iṉmai\VB oṭu\CL
10	காமனது kāmaṉatu	காமன்\VB அது kāmaṉ\VB atu

9. CONCLUSIONS

This paper has described the morphological analyzer based on the new and state of the art machine learning approaches. This paper demonstrated a new methodology adopted for the morphological analyzer for Classical Tamil texts. The rule based approach concludes that the Morphological Analyzer is the most important activity of any Natural Language Applications. The role and accuracy of any NLP applications development dependent on the accuracy of Morphological Analyzer. The rule based approach produce the best accuracy of tagged corpus in Classical Tamil texts, based on the 93 rules have been implemented in this analyzer its produce the 83% of results and identified the un-analyzed words in Classical Tamil machine learning rules have to developed. If unable to analyze the words in the rules it needs to go for the semantic rules, after developing the machine learned semantic rules the analyzer produce the improved results.

REFERENCES

[1] Anand Kumar M, Dhanalakshmi V, Soman K.P, 2010 "A Sequence Labeling Approach to Morphological Analyzer for Tamil Language", International Journal on Computer Science and Engineering Vol. 02, No. 06

[2] Shlomo Yona, November. 2004. A Finite-state based morphological analyzer for Hebrew. Faculty of Social Science Department of Computer Science, University of Haifa.

[3] David W. Packrd. Computer-Assisted Morphological Analysis of Ancient Greek. At Innovative Projects in University Instruction, University of California.

[4] http://www.logiclaw.co.uk/BOOKS/faust/faust/www.levity.com/alchemy/latin/latintrans.html

[5] Girish Nath Jha, Muktanand Agrawal, Subash, Sudhir K. Mishra, Diwakar Mani, Diwakar Mishra, Manji Bhadra, Surjit K. Singh. “Inflectional Morphology Analyzer for Sanskrit”. At Special Centre for Sanskrit Studies, Jawaharlal Nehru University, New Delhi, India.

[6] Huihsin Tseng and Ken-Jiann Chen. 2002. “Design of Chinese Morphological Analyzer” In: proceedings of First SINGHAN Workshop.

[7] San Diego. Linguistics Department, University of California, USA and Karine Megerdoomian, Inxight Software “Finite-State Morphological Analysis of Persian”.

[8] A reference Grammar of Classical Tamil poetry by V.S. Rajam. 1992.

[9] Ankita Agarwal, Pramila, Shashi Pal Singh, Ajai Kumar, Hemant Darbari. 2014. “Morphological Analyser for Hindi - A Rule Based Implementation”. International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970).

[10] Andronov M. 1969. The Standard Grammar of Modern and Classical Tamil. Madras: New Century Book House, Pvt. Ltd.

[11] Anandan, P. K. Saravanan, Ranjani Parthasarathi and T. V. Geetha. 2002. “Morphological Analyzer for Tamil” International Conference on Natural language Processing.

[12] Programming C# 3.0, Jesse Liberty and Donald Xie, O’Reilly, 2007 5th Edition.

[13] Programming C#, 2002 by Jesse Liberty publisher: O'Reilly, 2nd Edition.

[14] Dr. K. Umaraj, 2009, “Electronic Dictionary for Sangam Literature” in the 8th Tamil International Internet Conference in the University of Cologne, Germany and International forum for Information Technology in Tamil, held at University of Cologne, Germany Volume 8: 1, October

[15] S. Rajendran. 2006. “Parsing In Tamil - Present State of Art”. In journals of Language in India. Vol. 6: 8, August.

[16] http://en.wikipedia.org/wiki/Unicode

[17] Koskenniemi .K, 1983, “Two –Level Morphology: A general Computational; Model for Word Recognition and Production”, University of Helsinki, Helsinki,

[18] Ahmed, S.Bapi Raju, Pammi V.S. Chandrasekhar, M.Krishna Prasad, 2002, “Application of Multilayer Perception Network for Tagging Parts-Of-Speech” Language Engineering Conference, University of Hyderabad, India,

[19] Khaled Shaalan. 2010. Rule-based Approach in Arabic Natural Language Processing. In International Journal on Information and Communication Technologies.

[20] Parameshwari K, 2011, “An Implementation of APERTIUM Morphological Analyzer and Generator for Tamil”, an e journal of Language in India (www.languageinindia.com) , May 2011 Special Volume: Problems of Parsing in Indian Languages,