A Rule based Kannada Morphological Analyzer and Generator using Finite State Transducer

Ramasamy Veerappan, Antony P J, S Saravanan
Research Scholar
Computational Engineering and Networking Centre,
Amrita Vishwa Vidyapeetham University, Coimbatore,
Tamil Nadu, India

Dr. Soman K P
Professor and Head
Computational Engineering and Networking Centre,
Amrita Vishwa Vidyapeetham University, Coimbatore,
Tamil Nadu, India

ABSTRACT
Morphology plays an essential role in machine translation and many other natural language processing applications. Developing a well fledged morphological analyzer and generator (MAG) tools for highly agglutinative language like Kannada is a challenging task. The function of morphological analyzer is to return all the morphemes and their grammatical categories associated with a particular word form. For a given root word and grammatical information, morphological generator will generate the particular word form of that word. In the proposed project, we have developed a rule based MAG using finite state transducer. This project has been developed as part of the development of a machine translation system for English to Kannada language. The performance of the system was tested randomly against a set of lexicon containing approximately twenty thousand root words including noun, verb, adjectives and adverbs.

Keywords
Morphology, Kannada, Finite state transducer, Agglutinative, Orthographic rules

1. INTRODUCTION
The morphological structure of an agglutinative language is unique and capturing its complexity in a machine analyzable and generatable form is a challenging job. Analyzing the internal structure of a particular word is an important intermediate stage in many natural language processing applications especially in bilingual and multilingual MT system. A Morphological analyzer is used to analyze the internal structure of the words of a language. On the other hand a morphological generator does exactly the reverse of it i.e. given a root word and grammatical information morphological generator will generate the particular word form of that root word. The role of morphology is very significant in the field of NLP, as seen in applications like MT, question-answering (QA) system, IE, IR, spell checker, lexicography etc. So from a serious computational perspective the creation and availability of a morphological analyzer for a language is important.

Kannada is one of the four major Dravidian languages of South India. It is a state language of Karnataka and is spoken by about 20 million people. It has a long linguistic of about 1,500 years and had a continuous literature for over 1,200 years. Kannada is a morphologically rich language in which morphemes combine with the root words in the form of suffixes. Even though Kannada is historically and linguistically rich, the development in natural language processing for Kannada is very slow. The main reasons includes: non-availability of large scale data resources and also due to the inherent complexities of the language.

To build a MAG for a language one has to take care of the morphological peculiarities of that language, specifically in case of machine translation. Some peculiarities of Kannada language such as, the usage of classifiers, excessive presence of vowel harmony etc. make it morphologically complex and thus, a challenge in natural language generation (NLG).

Generally there are two approaches used to develop morphological analyzer and generator. The first approach is called corpus based approach where a large sized well generated corpus is required for training using a machine learning algorithm. The performance of the system will depends on the feature and size of the corpus. The disadvantage is that corpus creation is a time consuming process. On the other hand, rule based approaches are based on a set of rules and dictionary that contains root and morphemes. In rule based approaches every rule depends on the previous rule. So if one rule fails, it will affect the entire rules that follow it. When a word is given as an input to the morphological analyzer and if the corresponding morphemes are missing in the dictionary then the rule based system fails [1]. This paper is about the design and development of MAG for Kannada language using the rule based approach by considering all the peculiarities. We have implemented the system using AT &T Finite State Machine.

The function of morphological analyzer is to segment the given word into component morphemes and assigning correct morpho-syntactic information. The table 1 shows examples for morphological analysis of Kannada words.

Table 1. Input/Output examples for morphological analyzer

Input	Output
ಆನೆಗಳು (AnegaLu)	ಆನೆ+ಗಳು (Ane+gaLu)
ಹೋಗುತ್ತೇನೆ (hOguttEne)	ಹೋಗು+ಉತ್ತ್+ಏನೆ (hOgu+utt+Ene)

The function of morphological generator is to combine the constituent morphemes to get the actual word. The table 2 shows examples for morphological generation of Kannada words.

Table 2. Input/Output examples for morphological generator

Input	Output
ಆನೆ+ಗಳು (Ane+gaLu)	ಆನೆಗಳು (AnegaLu)
ಹೋಗು+ಉತ್ತ್+ಇನೆ (hOgu+utt+Ene)	ಹೋಗುತ್ತೇನೆ (hOguttEne)

2. LITERATURE SURVEY

In general there are several approaches attempted for developing morphological analyzer. In 1983 Kimmo Koskenniemi developed a two-level morphology approach, where he tested this formalism for Finnish language [2]. In this two level representation, the surface level is to describe word form as they occur in written text and the lexical level is to encode lexical units such as stem and suffixes. In 1984 the same formalism was extended in other languages such as Arabic, Dutch, English, French, German, Italian, Japanese, Portuguese, Swedish, Turkish and developed morphological analyzers successfully. In the same time a rule based heuristic analyzer for Finnish nominal and verb forms was developed by Jappinen [3]. In 1996, Beesley developed an Arabic finite state transducer for MA using Xerox finite state transducer (XFST), by reworking extensively on the lexicon and rules in the Kimmo-style [4]. At 2000, Agirve introduced a word–grammar based morphological analyzer using the two- level and a unification- based formalism for a highly agglutinative language called Basque [5]. Similarly using XFST, karine made a Persian MA in 2004 and Wintner came up with a morphological analyzer for Hebrew in 2005 [6, 7]. Oflazer Kamel developed a Finite State Machine (FSM) based Turkish morphological analyzer. In 2008, using the syllables and utilizing the surface level clues, the features present in a word are identified for Swahili (or Kiswahili) language by Robert Elwell.

In case of Indian languages, AU-KBC Research Centre of Anna University developed a finite state automata based morphological analyzer for Tamil language [8]. Dr. Shailly Goyal and Dr. Niladri Chatterjee of Indian Institute of Technology Delhi, worked on Hindi noun phrase morphology for developing a link grammar based parser [9]. Mrs. Rita Mathu , Dr. Madhavi Sinha and Prof. Rekha Govil also worked on Hindi Morphology. Many attempts have been done in case of Bengali and Marathi language morphology. In Bengali, unsupervised methodology is used for developing a morphological analyzer and two-level morphology approach was used to handle Bengali compound words by Sajib Dasgupta, in 2007 [10]. Manish Shrivastav, Nitin Agrawal, Bibhuti Mahapatra, Smriti Singh and Pushpak Bhattacharyya worked on morphology based natural language processing tools for Indian languages. A morphological analyzer and generators for Telugu, Tamil and Kannada was developed by University of Hyderabad [2]. Rule based morphological analyzer have been developed for Sanskrit and Oriya by Girish Nath jha and Mohanty respectively.

We have made a literature survey on Kannada natural language processing and found the following developments: A Kannada indexing software prototype is developed by Settar in 2002 [11]. A Kannada Word net is attempted by Sahoo and Vidyasagar of Indian Institute of Technology, Bangalore, in 2003 [12]. T. N. Vikram and Shalini R Urs developed a prototype of morphological analyzer for Kannada language (2007) based on Finite State Machine [13]. This is just a prototype and does not handle compound formation morphology and can handle maximum 500 distinct nouns and verbs. A Paradigm based Morphological Analyzer for Kannada Language Using Machine Learning Approach was developed by Antony P J and Dr Soman K P of Amrita Vishwa Vidyapeetham in 2010 [14]. This is a morphological analyzer for Kannada verbs and can also handle compound verb morphology. Uma Maheshwar Rao G and Parameshwari K of CALTS, University of Hyderabad attempted to develop a morphological analyzer and generators for South Dravidian languages in 2010 [15]. A network and process model for Kannada morphological analysis/ generation was developed by K. Narayana Murthy and the performance of the system is 60 to 70% on general texts [16]. Recently (Jan- 2011) Shambhavi B. R and Dr. Ramakanth Kumar of RV College, Bangalore developed a paradigm based morphological generator and analyzer using a trie based data strucure [17]. The disadvantage of trie is that it consumes more memory as each node can have at most 'y' children, where y is the alphabet count of the language. As a result it can handle up to maximum 3700 root words and around 88K inflected words.

3. CHALLENGES IN KANNADA MORPHOLOGY

Kannada is a verb-final inflectional language with a relatively free word order. Kannada morphology is characterized as agglutinative or concatinative, i.e., words are formed by adding suffixes to the root word in a series. Most of the words may change spelling when stems are inflected. Normally root word is affixed with several morphemes to generate thousands of word forms. The complexity of developing MAG for Dravidian language like Kannada is comparatively higher than the other languages like English. Most of the words may change spelling when stems are inflected. In agglutinative language like Kannada normally root word is affixed with several morphemes to generate thousands of word forms. To build an effective morphological analyzer one should carefully analyze and identify all these roots and morphemes.

Due to the highly agglutinating nature of the Kannada language and the morphophonemic variations that take place at the point of agglutination, it is very difficult to mark word boundaries [14]. Design should possibly cover all types of inflections. For example, the different meaningful parts of the word 'ಒದಿಕೊಂಡಿದ್ದವನ' (OdikoMDiddavana) -> 'the one (masculine) who was reading' is:

ಒದು + ಇ + ಕೊಳ್ಳು +ಂಡ್ + ಉ + ಇರು + ದ್ದ್ + ಅ + ಅವನು + ಅ

Odu + i + koLLu +MD+ u + iru + dd + a + avanu + a

Root + VBP+ AUXV +PST+ VBP + AUXV + PST+ RP + PRON-3SM + ACC

3.1 Types and Features of Kannada Words

In general, there are three types of Kannada Words namely: i) namapada (Declinable words or nouns) ii) kriyapada (Conjugable words or Verbs) and iii) avyaya (Uninflected words). Nouns, Pronouns and Adjectives are belongs to declinable words and are inflected to differences of case, number and gender. Conjugable words are inflected to mark differences of person, gender, number, aspect, mood and tense. All the Kannada words are of three genders: masculine, feminine and neuter. Declinable and Conjugable words have two numbers: singular and plural. The singular has no particular distinguishing marker added. The plural marker is usually "gaLu", but there are some exceptions as

follows: Masculine nouns (E.g., huDuga) ending in "a" and some feminine nouns (E.g.,hemgasu) endings in "u" have plural with "aru" . Feminine nouns ending with "i (E.g.,huDugi)" or "e (atte)" have plural with "yaru". Also nouns with kinship terms (E.g., aNNa), the marker for plural is often "aMdiru". Some nouns are irregular plurals such as "makkaLu" which is the plural for noun "magu".

3.2 Noun Cases and Characteristics suffixes

The case system of Kannada is similar to those of other south Dravidian languages like Tamil, Telugu and Malayalam. Nouns may usually end in a, e, i, u, A, or in a consonant [18]. Various suffixes are added to the noun stem to indicate different relationships between the noun and other constituents of the sentence. The different types of suffixes are used with a particular case based on the type of nouns and their end character. For example "dative" case characteristic suffixes are decided by the following criteria as shown in table 3.

Table 3. Dative Case Characteristics suffixes for Nouns

Noun type	Ends with	Dative suffix	Example noun	Dative form
Neuter noun	ಅ (a)	ಕ್ಕೆ (kke)	ಮರ (mara)	ಮರಕ್ಕೆ (marakke)
	ಎ,ಇ,ಉ (e,i,u)	ಗೆ (ge)	ಮನೆ (mane)	ಮನೆಗೆ (manege)
	consonants	ಇಗೆ (ige)	ಊರು (Uru)	ಊರಿಗೆ (Urige)
Neuter determinative	-	ಅಕ್ಕೆ (akke)	ಇದು (idu)	ಇದಕ್ಕೆ (idakke)
Rational noun	-	ಇಗೆ (nige)	ಅಣ್ಣ (aNNa)	ಅಣ್ಣನಿಗೆ (aNNanige)

Table 4 below shows the different cases and their corresponding characteristic suffixes for nouns.

Table 4. Noun Cases and their Characteristics suffixes

Feature	Characteristic Suffix	
	Singular	Plural
Nominative (Prathama)	ವು(vu)/ಯು(yu)/ಉ(u)/ನು(nu)	ಗಳು(gaLu)/ರು(ru)/ಂಗಿರು(Mdiru)/ಯರು(yaru)
Accusative (Dwitiya)	ವನ್ನು(vannu)/ಯನ್ನು(yannu)/ಅನ್ನು (annu)/ನನ್ನು (nannu)	ಗಳನ್ನು(gaLannu)/ಂಗಿರನ್ನು (Mdirannu)/ಯರನ್ನು(yarannu)/ರನ್ನು(rannu)
Instrumental (Tritiya)	ದಿಂದ(diMda)/ ಯಿಂದ (yiMda)/ ಇಂದ(iMda)/ ನಿಂದ(niMda)	ಗಳಿಂದ(gaLiMda)/ಂಗಿರಿಂದ(MdiriMda)/ಯರಿಂದ(yariMda)/ರಿಂದ(riMda)
Dative (Chaturthi)	ಕ್ಕೆ(kke)/ಗೆ(ge)/ಇಗೆ (ige)/ನಿಗೆ(nige)/ಅಕ್ಕೆ (akke)	ಗಳಿಗೆ(gaLige)/ಂಗಿರಿಗೆ(Mdirige)/ಯರಿಗೆ(yarige)/ರಿಗೆ(rige)
Ablative (Pachami)	ದಿಂದ(deseyiMda)	ಗಳದಿಂದ(gaLadeseyiMda)/ಂಗಿರದಿಂದ(Mdiradeseyida)/ಯರದಿಂದ(yaradeseyiMda)/ರದಿಂದ(radeseyiMda)
Genitive (Shashti)	ದ(da)/ಯ(ya)/ಇನ(ina)/ನ(na)/ಅ(a)/ವಿನ(vina)/ಅರ(ra)	ಗಳ(gala)/ಂಗಿರ(Mdira)/ಯರ(yara)/ರ(ra)
Locative (Saptami)	ದಲಿ,,(dalli)/ಯಲಿ,,(yalli)/ಅಲಿ,,(alli)/ ನಲಿ,,(nalli)	ಗಳಲಿ,,(gaLalli)/ಂಗಿರಲಿ,,(Mdiralli)/ಯರಲಿ,,(yaralli)/ರಲಿ,,(ralli)
Vocative (Sambhodana)	ಏ(E)/ಎೇ(vE)/ಅ(A)/ಈ(I)	ಗಳೇ(gale)/ಂಗಿರೇ(MdirE)/ /ಯರೇ(yare)

3.3 Verb Morphology

Comparing with other Dravidian language like Malayalam, the morphological structure of Kannada is more complex because it inflects to person, gender, and number markings [14]. In case of verb morphology each root word is combines with auxiliaries that indicate aspect, mood, causation, attitude etc. The uniqueness in the structure of verbal complexity makes it very challenging to capture in a machine analyzable and generatable format. Also the formation of the verbal complex involves arrangement of the verbal units and the interpretation of their combinatory meaning. Phonology also plays a little role in word formation in terms of 'morphophonemic' and 'sandhi' rules which account for the shape changes due to inflection.

Verb forms can be broadly classified into two types: finite verbs and non-finite verbs. In case of finite verbs, the verbs are usually added to the end of sentences with the exception of Clitics and can have nothing added to them. The general syntax of finite verb is the form: Subject-Object-Verb. Some of the finite forms of the verbs are imperatives, present and past forms marked with PNG, modals and verbal/participle nouns. The tense can be past/present/future, if it is in the affirmative. The negative form does not take tense. The non-finite verbs in contrast cannot stand alone and must have some other forms following them. Non-finite verb forms include infinitives, verbal and adjectival participles and tense-marked verb stems [19]. The non-past denotes both present and future tenses and unlike Malayalam language (another south Dravidian language) all tenses have different tense markers in Kannada language. Mood is another important feature of Kannada language and is associated with statements of fact versus possibility, supposition, etc [20]. There are four different moods that are expressed in Kannada are: infinitive, imperative, affirmative and negative. Also Kannada has some additional modal forms such as: indicative, conditional, optative, potential, monitory and conjunctive.

Kannada language also include past verb stems in addition to simple verb stems, that are used in forming the past tense, past participles, conditionals and some other constructions. The past

stems also form the base to which contingent PNG markers are added. The contingent form is another distinguished feature of Kannada language that is not present in any other Dravidian languages [21]. Table 5 shows the features of Kannada words with examples.

Table 5. Verb features and Characteristics suffixes

Feature	Characteristic Suffixes	Example
Infinitive	'ಅಲ್ 'ಒಕ್ಕೆ' (Okke).	ಬರು (baru) -> come + ಅಲ್ (al) + ಇಲ್ಲ(illa) -> negative = ಬರಲಿಲ್ಲ (baralilla) -> didn't come
Imperative	O(yO)/ E(yO)/ iri(yiri)	ಹೋಗೋ (hOgO)/ ಹೋಗೇ (hOgE/ಹೋಗಿರಿ (hOgiri)
Negative Imperative	'ಬಾರದು' (bAradu) / 'ಬೇಡ' (bEDa)/ 'ಕೂಡದು' (kUDadu)	ಹೋಗು (hOgu) + ಅ (a) + ಬಾರದು (bAradu) = ಹೋಗಬಾರದು (hOgabAradu) -> 'don't go'
Optative	'ಇ' (i)	ಮಾಡು (maDu) + ಅಲ್ (al) + ಇ (i) = ಮಾಡಲಿ (mADali), 'let do'
Hortative	'ಒಣ' (ONa)	ಮಾಡು (mADu)+ ಒಣ (ONa) = ಮಾಡೋಣ (mADONa), 'let's do'
Participle:	'ಅ' (A)/ 'ಇ' (i)/ 'ಅದೆ'(ade)/'ಅದು' (adu)/'ಅದ' (ada)	ನೋಡು + ಅದೆ = ನೋಡದೆ (nODu + ade = nODade) -> 'without seeing'
Verbal aspect markers	'ಬಿಡು'(biDu)/' ಹೋಗು'(hOgu)	ಬಿಟ್ಟು ಬಿಡು (biTTu biDu) 'let go'
causative suffix	'ಇಸು'(isu) / 'yisu'	'ಕಲಿ' (kali -> learn) + 'ಇಸು'(isu) -> 'ಕಲಿಸು' (kalisu -> teach)
conditional suffix	'are'	'ಹೋದರೆ' (hOdare) 'if (someone) goes, (then…)'

The Person–Noun-Gender (PNG) and the tense marker concatenated to the verb stems are the two important aspect of verb morphology [14]. The verbal inflectional morphemes attach to the verbs providing information about the syntactic aspects like number, person, case-ending relation and tense. Usually the Kannada verbs follow the regular pattern of suffixation. The table 6 shows the various PNG suffixes that can be attached to be any verb root word.

Table 6. Kannada PNG- Suffixes

Person	Number	Gender	PNG Suffix			
			Present	Future	Past	Contingent
First	Singular	Masculine/ Feminine	ಏನೆ (Ene)	ಏನು, ಎ (enu, e)	ಏನು, ಎ (enu, e)	ಏನು (Enu)
	Plural	Masculine/ Feminine	ಏವೆ (Eve)	ಏವು (evu)	ಏವು (evu)	ಏವು (Evu)
Second	Singular	Masculine/ Feminine	ಈ, ಈಯೆ (I, Iye)	ಇ, ಇಯೆ (i, iye)	ಇ, ಇಯೆ (i, iye)	ಈಯ (Izha)
	Plural	Masculine/ Feminine	ಈರಿ (Iri)	ಇರಿ (iri)	ಇರಿ (iri)	ಈರಿ (Iri)
Third	Singular	Masculine	ಆನೆ (Ane)	ಆನು (anu)	ಆನು (anu)	ಆನು (Anu)
	Singular	Feminine	ಆಳೆ (Ale)	ಆಳು (aLu)	ಆಳು (aLu)	ಆಳು (ALu)
	Plural	Masculine/ Feminine	ಆರೆ (Are)	ಆರು (aru)	ಆರು (aru)	ಆರು (Aru)
	Singular	Neuter	ಇದೆ (ide)	ಉದು (udu)	ಇತ್ತು, (ittu)	ಈತ್ತು (Ittu)
	Plural	Neuter	ಇವೆ (ive)	ಅವು (avu)	ಅವು (avu)	ಆವು (Avu)

4. IMPLEMENTATION OF MAG MODEL

The proposed rule based MAG tool was developed using AT &T Finite State Machine. This section explains the various efforts required to create the proposed MAG system.

4.1 Classifying Verb Paradigms

One of the most important steps involved in the creation of MAG is to classify the verb paradigms with computational perspective. Most of the cases the problem arises due to past tense markers that change from one paradigm to another [22]. Past verbs are broadly classified into two types called regular and irregular (or semi regular). In case of regular the different words are formed by adding 'id' to the verb stem. In the other case different words are formed by adding any one of the past tense marker as shown in table 7. To resolve the computational challenges in verb morphological analysis we have classified verbs into 35 distinguished paradigms and verb words are grouped based on their class paradigms [14].

Table 7. Proposed Kannada Verb Paradigms

Paradigms	Past tense marker	Description & Example
Class-1	-ತ್ತ-(-tt-)	Verbs ends with 'Ayu', 'Iyu', 'ILu' Eg: sAyu, Iyu, kILu etc.
Class-2	-ತ್ತ-(-tt-)	Verbs ends with 'eru', 'aLu', 'uLu' Eg: 'heru','horu','aLu',,'uLu' etc.
Class-3	-ತ್ತ-(-tt-)	Verbs ends with 'aLu','uLu' Eg : aLu, uLu
Class-4	-ನ್ತ- (-Mt-)	Verbs ends with 'illu' Eg : nillu
Class-5	-ತ್- (-t-)	Verbs ending with 'I' and 'e' Eg: 'kali', 'bali', 'mere', 'koLe' etc.
Class-6	-ತ್- (-t-)	Verbs ends with 'ULu' Eg: Example: hULu
Class-7	-ತ್- (-t-)	Verbs ends with 'Olu', 'Ulu', 'Elu' Eg: 'jOlu','sOlu','nUlu','hElu' etc.
Class-8	-ದ್- (-d-)	Verbs ending with 'Ayu','Oyu','Eyu','Iyu' Eg:' kAyu', 'kOyu' ,'tEyu','sIyu','hAyu' etc.
Class-9	-ದ್- (-d-)	Verbs ending with 'Agu','Ogu' Eg: 'hOgu', 'Agu' etc
Class-10	-ದ್- (-d-)	Verbs ends with 'are' Eg: 'bare'
Class-11	-ದ್- (-d-)	verbs ending with 'ge' and 'gi' Eg: 'age', 'agi'
Class-12	-ದ್- (-d-)	Verbs ending with 'yyu' Eg: 'koyyu', 'geyyu', 'hoyyu', 'bayyu', 'suyyu' etc.
Class-13	-ದ್- (-d-)	Verbs ends with 'nnu' Eg:' annu', 'tinnu', 'ennu' etc
Class-14	-ದ್- (-d-)	Verbs ending with 'Eyu' Eg: 'gEyu', 'nEyu' etc
Class-15	-ದ್- (-d-)	Verbs ending with 'Ayu' Eg: 'Ayu'
Class-16	-ದ್ದ-(-dd-)	Verbs ends with 'iru' Eg: 'iru'
Class-17	-ದ್ದ-(-dd-)	Verbs ends with 'kaLu' Eg: kaLu
Class-18	-ದ್ದ-(-dd-)	Verbs ends with 'ILu','ELu' Eg: 'bILu' ,'ELu', etc
Class-19	-ದ್ದ-(-dd-)	Verbs ends with 'Eyu' Eg: :mEyu
Class-20	-ದ್ದ-(-dd-)	Verbs ends with 'ellu' Eg: 'gellu'
Class-21	-ದ್ದ-(-dd-)	Verbs ends with 'ADu', 'ODu' Eg: 'ADu','nODu', 'kADu', 'tODu' , etc
Class-22	-ಇದ್- (-id-)	Verb ends with 'TTu','ddu', 'bbu','ttu', 'llu', 'ccu' Eg: aTTu, addu, ubbu, kuttu, cellu, heTTu, beccu,hottu etc
Class-23	-ಇದ್- (-id-)	verbs ending with 'Oru', 'Eru' Eg: tOru, sEru, hEru ,hOru etc
Class-24	-ಇದ್- (-id-)	Verbs ends with 'ju','Du','su' Eg: mOju, ADu, aMkurisu etc
Class-25	-ಇದ್- (-id-)	Verb ends with 'MTu',Mju, Mcu Eg. IMTu, aMju, hoMju etc
Class-26	-ಇದ್- (-id-)	Verbs ends with 'ELu', 'ILu' Eg: hELu, sILu etc
Class-27	-ನ್ದ- (-nd-)	Verbs ends with 'Eyu', 'Oyu' Eg: bEyu, nOyu etc
Class-28	-ನ್ದ- (-nd-)	Verbs ends with 'A'(aru) Eg: taru(tA),baru(bA) etc.
Class-29	-ನ್ದ- (-nd-)	Verbs ends with 'ollu', 'ellu', 'allu' Eg: kollu,mellu ,sallu etc.
Class-30	-ನ್ನ- (nD-)	Verb stems ending with 'ANu' Eg: kANu
Class-31	-ನ್ನ- (nD-)	Verb ends with 'oLLu' Eg: koLLu
Class-32	-ಟ್- (-T-)	Verb ends with 'aDu','eDu', 'oDu','iDu','uDu' Eg: aDu, keDu, koDu, naDu, iDu, uDu, toDu, paDu, haDu etc
Class-33	-ಕ್- (-k-)	Verb ends with 'ggu'and "gu' Eg: oggu, miggu(migu), hoggu, sigu, nagu, etc
Class-34	-ದ್- (-d-)	Verbs ends with 'kAyu' Eg: : kAyu, dArikAyu
Class-35	-ನ್ನ- (nD-)	Verbs ends with 'kAyu' Eg: baggiko, bEDiko etc

4.2 Information required to build MAG

The following information's are required to build a morphological analyzer and generator:

4.2.1 Lexicon

The list of stems and affixes together with basic information's about them (Noun stem or Verb stem etc,).

4.2.2 Morphotactic

The model of morpheme ordering that explains which classes of morphemes can follow other classes of morphemes inside a word. E.g., the rule that Kannada plural morpheme follows the noun stem rather than preceding it.

4.2.3 Orthographic rules

These are spelling rules used to model the changes that occur in a word, usually when two morphemes combine. For example, insert a "yu" on the surface tape just when the lexical tape has a morpheme ending in 'e' (or i, etc) and the next morphemes are "tt"(PRES) and "Ane"(3SM).

beLe + insert“yu” + PRES(tt) + 3SM(anu) ->beLe-yu-tt-Ane =beLeyuttAne

4.3 Creation of rules using FST

The proposed rule based MAG tool was developed using AT &T Finite State Machine (FST). A finite state transducer essentially is a finite state automaton that works on two (or more) tapes. The most common way to think about transducers is as a kind of “translating machine” which works by reading from one tape and writing onto the other. For example, on one tape we read “ಉದ್ಯಾನಗಳು”, on the other we write “ಉದ್ಯಾನ +N +PL”, or the other way around as shown in figure 1. “ಉ : ಉ” means read a “ಉ” symbol on one tape and write the same “ಉ” on the other tape. Similarly “+N:ε” means read a “+N” symbol on one tape and write nothing on the other.

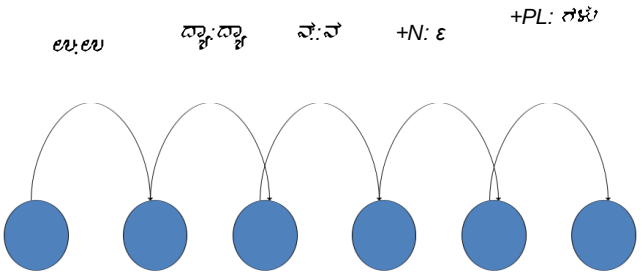

Fig 1. FST working principle

FST’s can be used for both analysis and generation (they are bidirectional) and it act as two level morphology as shown in figure 2 [23]. Represent a word as a correspondence between a lexical level and surface level. At lexical level represents a simple concatenation of morphemes making up a word. But at the surface level represents the actual spelling of the final word.

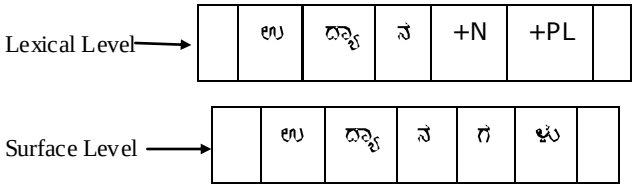

Fig 2. FST as Two-level morphology

4.4 Architecture of Proposed MAG Model

With all relevant morphological feature information of Kannada words we have created well defined sandhi rules based on finite state transducer. The architecture of proposed a MAG tool is as shown in figure 3.

The system is based on lexicon and orthographic rules from a two level morphological system. For the Morphological generator, if the string which has the root word and its morphemic information is accepted by the automaton, then it generates the corresponding root word and morpheme units in the first level as shown in figure 4.

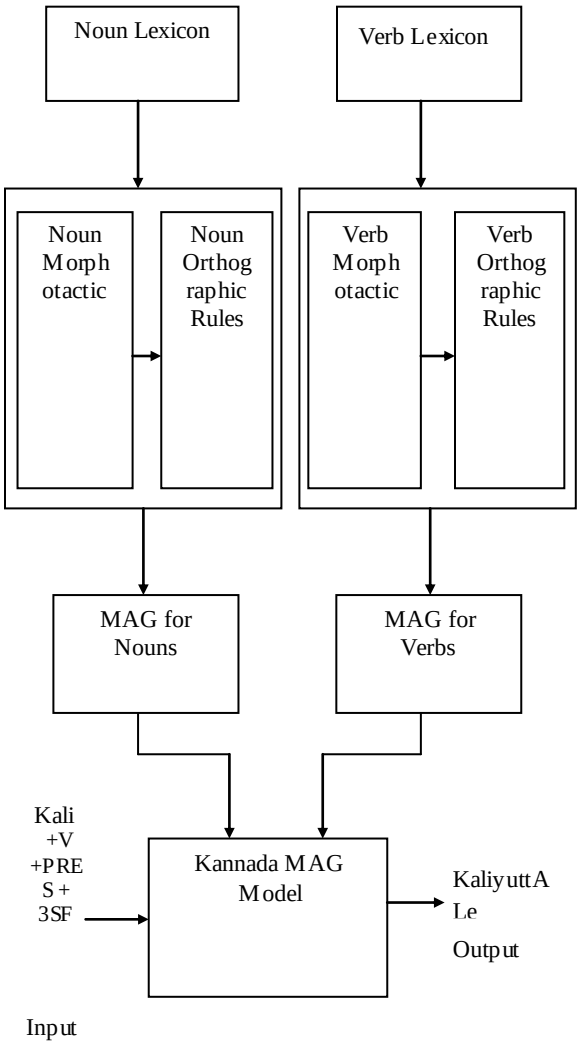

Fig 3. Architecture of proposed MAG model.

Here “beLe” is the root word, “V” indicates the category of the root word as verb, “PRES and FUT” indicates the tense markers for presentence and future tense respectively and 3SM indicates PNG marker for third singular masculine.

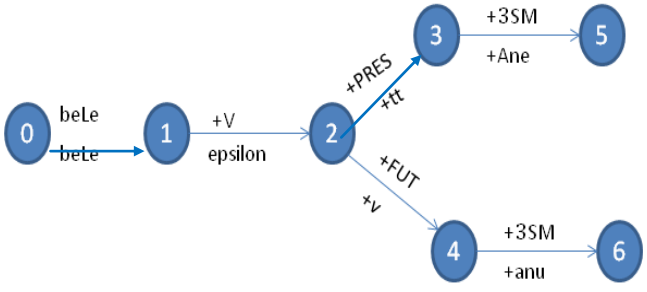

Fig 4. Example for Morphotactics Rule

The output of the first level becomes the input of the second level where the orthographic (sandhi) rules are handled as shown in Figure5. If it gets accepted then it generates the inflected word.

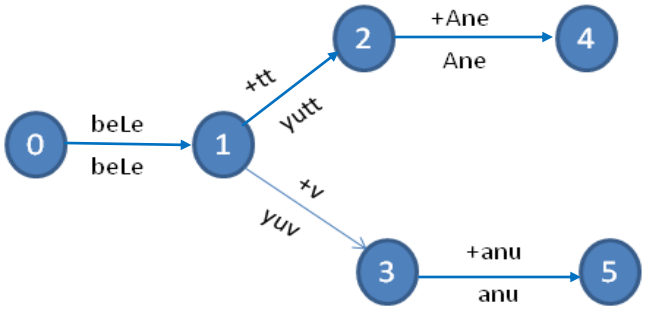

Fig 5. Application of Sandhi Rule

The sandhi rule should be written in such a way that, if the root word ends with "e" and the next morphemes are "tt"(PRES) or "Ane"(3SM), then insert "yu" immediately after the root word. Figure 6 below shows the corresponding sandhi rule.

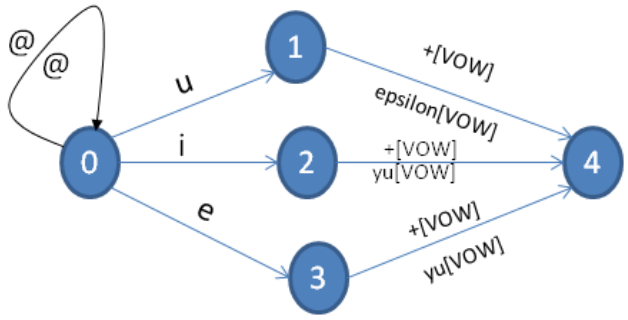

Fig 6. Example for Sandhi Rule

4.5 GUI of Proposed MAG Model

Sample screenshots of the proposed MAG model for noun are shown in figures 7 and 8. Similarly figures 9 and 10 shows the screenshots of the proposed MAG model for verb.

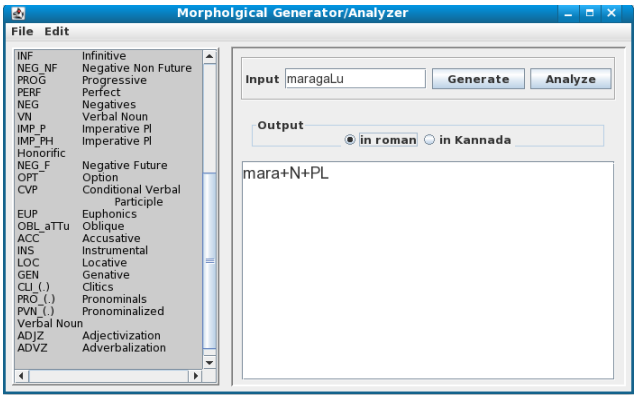

Fig 7. GUI of Kannada Morph analyzer for Noun

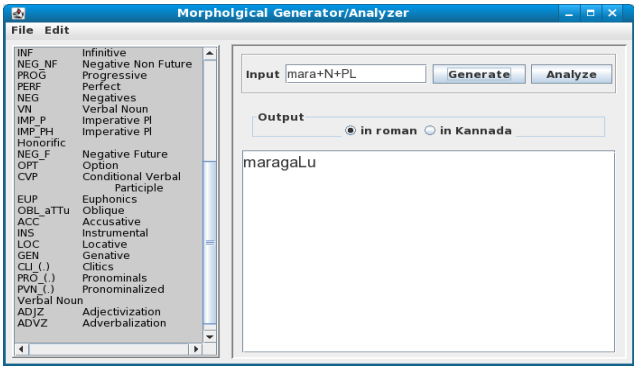

Fig 8. GUI of Kannada Morph generator for Noun

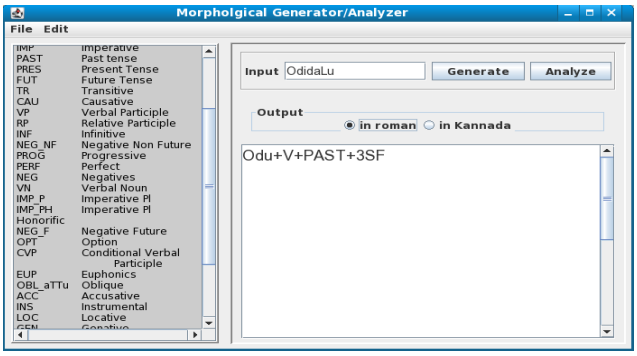

Fig 9. GUI of Kannada Morph analyzer for Verb

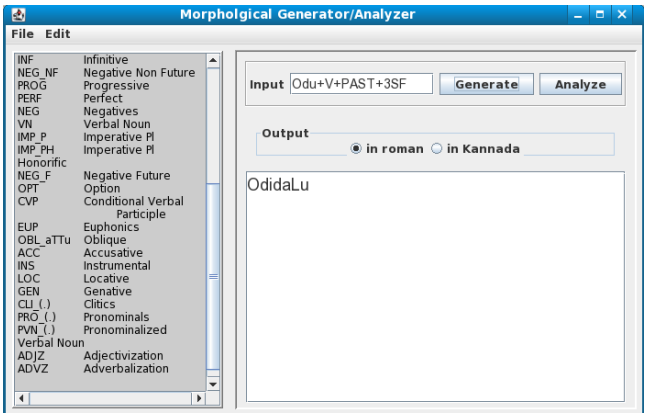

Fig 10. GUI of Kannada Morph generator for Verb

5. SYSTEM PERFORMANCE AND CONCLUSION

Development of MAG is a challenging task for all types of word forms. The proposed MAG is capable of analyzing and generating a list of twenty thousand nouns, around three thousand verbs and a relatively smaller list of adjectives. The uniqueness of the proposed MAG is its capacity to generate and analyze transitive, causative and tense forms apart from the passive constructions, auxiliaries and verbal nouns. The performance of the proposed system can be substantially improved by adding more rules such as rules for complex morphology etc. Also by checking against more and more different types of word lexicons, the accuracy of

the proposed MAG can be improved. A rule based machine translation system for English to Kannada language was developed using the proposed MAG.

6. ACKNOWLEDGMENTS

We acknowledge our sincere gratitude to Dr. Rajendran S (Tamil University, Tanjavur, Tamil Nadu, India) and Prof. M Shankaranarayana Bhat (Head of Kannada department and Principal, Junior College, Sampaje, Coorg, Karnataka, India) for their excellent support to generate Kannada word paradigms. We also express our gratitude to Mr. Harsha (Research Scholar, CEN, AMRITA Vishwa Vidyapeetham, Coimbatore, India) and Ms. Dhanya (CEN, AMRITA Vishwa Vidyapeetham, Coimbatore, India) for their valuable support and encouragement for developing the rule based Kannada MAG tool.

7. REFERENCES

[1] Dhanalakshmi V., Anand Kumar M., Rekha R.U., Arun Kumar C., Soman K.P., Rajendran S.:' Morphological Analyzer for Agglutinative Languages Using Machine Learning Approaches', Advances in Recent Technologies in Communication and Computing, International Conference on Advances in Recent Technologies in Communication and Computing, 2009.

[2] Uma Parameshwari Rao G, Parameshwari K: CALTS, University of Hyderabad, 'On the description of morphological data for morphological analyzers and generators: A case of Telugu, Tamil and Kannada'.

[3] Harri Jappinen, 'Knowledge engineering approach to morphological analysis', first conference on European chapter of the Association for Computational Linguistics.

[4] Beesley, K. and L. Karttunen. 'Finite State Morphology'. Stanford, CA: CSLI Publications, 2003.

[5] Aduriz I, Agirre E., 'A word-grammar based morphological analyzer for agglutinative languages', University of the Basque Country.

[6] Karine Megerdoomian 'Finite-State Morphological Analysis of Persian', Inxight Software, Inc, University of California, San Diego.

[7] Shuly Winter, 'Hebrew Computational Linguistics: Past and Future', Artificial Intelligence Review 21: 113–138, 2004, Kluwer Academic Publishers.

[8] nlp.au-kbc.org/ma_language_format_final.pdf

[9] Shailly Goyal, 'Parsing Aligned Parallel Corpus by Projecting Syntactic Relations from Annotated Source Corpus', Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, pages 301–308, Sydney, July 2006. Association for Computational Linguistics.

[10] Sajib Dasgupta, 'Morphological Analysis of Inflecting Compound Words in bangla', BRAC University, Dhaka, Bangladesh.

[11] S Settar, Sanjoy Goswami, H.K Abhishek, 'Indexing Software for Ancient Kannada Books' Proceeding LEC '02 Proceedings of the Language Engineering Conference (LEC'02).

[12] Sahoo k and Vidyasagar K V, 'Kannada Wordnet- A lexical database', TENHON 2003, Conference on convergent Technologies for Asia-Pacific Region.

[13] T.N. Vikram & Shalini R Urs, (2007), 'Development of Prototype Morphological Analyzer for the South Indian Language of Kannada', Lecture Notes In Computer Science: Proceedings of the 10th international conference on Asian digital libraries: looking back 10 years and forging new frontiers. Vol. 4822/2007, 109-116.

[14] Antony P J, M. Anand Kumar and K.P. Soman, 'Paradigm based Morphological Analyzer for Kannada Language Using Machine Learning Approach', International journal on Advances in Computational Sciences and Technology ISSN 0973-6107 Volume 3 Number 4 (2010) pp. 457–481.

[15] Language in India, www.languageinindia.com/may2011/ v11i5may2011.pdf.

[16] 202.41.85.68/knm-publications/morph-icosal2.pdf.

[17] Shambhavi. B. R, Dr. Ramakanth Kumar P, Srividya K, Jyothi B J, Spoorti Kundargi, Varsha Shastri G, 'Kannada Morphological Analyser and Generator Using Trie', International Journal of Computer Science and Network Security, VOL.11 No.1, January 2011.

[18] http//ccat.sas.upenn.edu/plc/kannada/grammar/KannadaChap .2.pdf.

[19] http//ccst.sas.upenn.edu/plc/kannada/grammar/KannadaChap .3.pdf.

[20] B.A Sharada, Transformation of Natural Language into Indexing Language: Kannada - A Case Study.

[21] Dr. K. Kushalappa Gouda: Kannada Sankshipta vyakarana, Kannada University, Hampi, Publication: Suvarna Karnataka, 2006.

[22] S.N. Sridar: "KANNADA", a Kannada grammar book, Series Editor, Bernard Comrie.

[23] Dr. A.G. Menon, S. Saravanan, R. Loganathan and Dr. K. Soman, Amrita University, Coimbatore, India. 'Amrita Morph Analyzer and Generator for Tamil: A Rule Based Approach'.