Turkish Journal of Computer and Mathematics Education Vol.12 No. 11 (2021), 2826- 2838
Research Article

Kannada Morphological Analyser and Generator Using Natural Language Processing and ML Approaches

AnithaG[a], G Sunil Kumar[b], Manjunath Swamy B E[c], Thriveni J[d], Venugopal K R[e]

[a] Department of Computer Science& Engineering, Vijaya Vittala Institute of Technology Bengaluru, Karnataka, India
[b,d] Department of Computer Science & Engineering, Bangalore University, UVCE Bengaluru, Karnataka, India
[c] Department of Computer Science & Engineering, Don Bosco Institute of Technology, Bengaluru, Karnataka, India
[e] Bangalore University, Bengaluru, Karnataka, India

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 10 May 2021

Abstract: The Morphological analyser & generator (MAG) plays an important Natural LanguageProcessing application and are compulsory apparatuses in Machine Translation. In the work proposed, MAG for Kannada language were created utilizing rule based just as factual methodology by fusing morphological data and characteristics of the language. The classification of words into various paradigms dependent on their inflections and orthographic changes, structuring the morphological framework, rule and corpus creations are the primary difficulties in the advancement of generator.The MAG tool is constructed for Kannada texts using machine learning strategies and produced theoutcomes using numerouskinds of eight input data sets. The validation of GUIs and processing of tool is carried out and the snapshots along with the tabulated and graphical performance analysis are published in the work.

Keywords: NLP, MT, Lexicon, Morphological Generator, Stemmer, SVM

1. Introduction

The NLP-Natural Language Processing is a field dealing with the computational aspects of the human language. The aim of Natural Language Processing is to break down & comprehend common dialects utilized by people & decode linguistic data to rules or portrayal of various types. The MLsystems have accomplished a major advantage overcomplicated linguistic language structure. NLP has advancement dramatically, using measurable strategies prepared on enormous labelled corpora. Factual labelling approaches can handle intrinsic uncertainty issue by appointing a likelihood to every term. Another drawback of factual calculations is that they needa large amount of explained training information.

For an agglutinative language like Kannada, improving a morphological analyser & generator for a broad range of word forms is a difficult task. The aim of the usage was to consolidate increasingly linguistic data in Kannada language as well as better semantic highlights, in order to address the morphological problem more effectively. Performance of the factual methodology relies upon enormous measured adjusted bilingual corpora of a wide range of word classifications. On the hand the exhibition of the standard methodology based with respect to a wide range of basic and complex semantic guidelines, so as to cover a broad range of term structures. By adding more guidelines and testing against an ever-increasing number of various kinds of lexicon words, the output of the developedrule-basedsystemmay be dramatically improved. Then again, presentation of factual methodology may be enhancedthrough expanding the size of the corpus to include other word classes such as adverb, pronoun, noun and so forth. The Morphological analyser depends on paradigm to become familiar with the action word types of Kannada language by fusing machine learning methodologies.

The work is to build up a framework which can distinguish Kannada words, examine different highlights of that word and create word for a given arrangement of highlights with the assistance of Machine Learning procedures utilizing SVM calculation with better prescient exactness.

The accessible morphological analysers for Kannada dialect follow stemming dependent, corpus dependent and paradigm-dependent methodologies. The corpus dependent methodology has the burden that the huge volume of information should be prepared. The stemming dependent methodology follows the procedure of isolating the word to its relating stems and the addition or prefix and afterward process this. To break down a word it must have the relating stems in its word reference. This methodology might effectively dissect standard words yet it can't break down sporadic words. The paradigm characterizes all the word types of a given stem and furthermore gives an element structure each word structure.

Over the most recent two decades, there has been an upset in the improvement of Indian characteristic language processing. Despite the fact that Kannada is a language wealthy in literature, its assets are poor when seen through the crystal of computational phonetics [3]. Kannada is an exceptionally agglutinative and morphologically rich dialect. Syntactic and semantic fluctuation makes the issue a lot harder for making very much fledged computational phonetic instruments and MT framework for Kannada language.The effect of the data innovation on NLP research has altered the method of how language was examined and comprehended. The

focal objective of this exploration work is to build up a standard based MT framework for English to Kannada language by incorporating various modules and different computational phonetic devices.

Machine Transliteration - English to Kannada: The linguistic interpretation from the local language (English) to unknown dialect (Kannada) is characterized as machine transliteration. English to Kannada Machine Transliteration device interprets specialized terms and legitimate names, for example, individual, area and association names from English language into Kannada language with inexact phonetic reciprocals. An English named element word can be converted into a few potential objective words and the transliteration plans to yield the specific objective word dependent on the way to express the objective language. For instance, the individual named "Akaar" could be converted into various objective Kannada words as demonstrated as follows:

AkAr (ಆಕಾರ್)→ right Pronunciation

akAr (ಅಕಾರ್)

Akar (ಆಕರ್)

akar (ಅಕರ್)

Tagging of Parts of Speech: The tagging of POS- Part of discourse is the way toward doling out the grammatical form tag or other lexical class marker to every single word in a sentence. For instance, the contribution to the POS tagger is a Kannada sentence "ರಾಜುಕಲ್ಲನ್ನುಎಸೆದನು". The tagger labels every single word in the given sentence input dependent on the proposed tagset by comprehending various ambiguities as demonstrated as follows.

ರಾಜು<NNP>ಕಲ್ಲನ್ನು<NN>ಎಸೆದನು<VF>. <DOT>

Kannada-MAG-Morphological Analyser and Generator: Kannada morphological analyserhas the ability to recover all morphemes & their syntactic groups associated with a particular Kannada word structure as shown below:

Input: ಆಸೆಗಳು(AsegaLu)

Output: ಆಸೆ + ಗಳು(Ase+gaLu)

Input: ಹೋರಾಡುತ್ತೇನೆ(hOraduttEne)

Output: ಹೋರಾಡು+ಉತ್ತ್+ಏನೆ

(hOradu+utt+Ene)

Then again, the capacity of morphological generator is simply converse of that analyser. The morphological generator will create the particular word form of that word for a given root word&linguistic information.

Syntactic Analysing for Sentence in Kannada: A syntactic parser instrument perceives a sentence in Kannada and doles out a syntactic structure to it. For instance, the contribution to the syntactic parser is a sentence in Kannada, ರಾಜುಕಲ್ಲನ್ನುಎಸೆದನು. ‟ Raju tossed the stone”. The syntactic parser perceives the sentence by illuminating lexical and connection ambiguities and doles out a syntactic structure to it as a parse tree as appeared in Fig.1

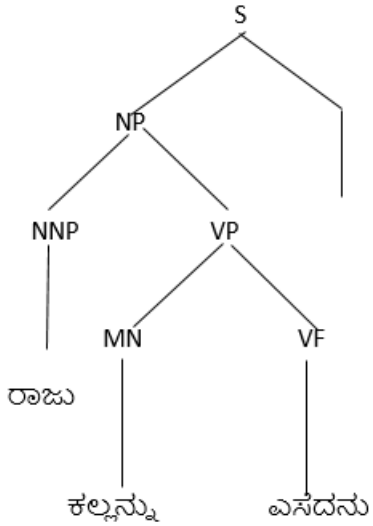


Fig. 1:The Syntactic structure analysing kannada sentence

The MT for English to Kannada is the use of PCs to the assignment of deciphering writings from English to Kannada language. For instance, for the information English sentence "Rama tossed the stone", the MT framework must deliver the proportional Kannada sentence as "ರಾಜುಕಲ್ಲನ್ನುಎಸೆದನು".

The Machine Translation is the utilization of PCs to the undertaking of deciphering writings starting with one common language then onto the next. MT is the one of the soonest application, that uses different degrees of NLP running from the "word-based‟ way to deal with applications that incorporate more significant levels of examination.

By and large, the NLP methodologies are sorted in various categories: statistical, symbolic, connectionist and hybrid. The initial two methodologies have existed together since from the earliest starting point of NLP. In 1960, connectionist methodology was presented in NLP and has created barely any frameworks dependent on this methodology. There after the methodologies like symbolic ruled the NLP field for quite a while. Because of the accessibility of basic computational assets and the need to manage wide, genuine settings, the factual methodologies recaptured its prevalence in the year 1980. Simultaneously connectionist approaches additionally recouped by showing the utility of neural systems in NLP.

Machine Learning: The Machine learning manages methods that permit PCs naturally figure out how to make exact forecasts dependent on past perceptions. The significant focal point of ML is to extricate data from information consequently, by computational and factual strategies. The ML strategies are being utilized for understanding numerous errands of NLP. They incorporate discourse acknowledgment, report classification, record division, grammatical feature labelling, and word-sense disambiguation, named substance acknowledgment, MT, parsing and transliteration.

The supervised learning is a programmed realizing, which centres around displaying input/yield connections. The objective of managed or supervised learning is to recognize an ideal practical mapping between the incoming info P, portraying the incoming design, to the yield variable, that is, a class name Q to such an extent that Q = f(P), that is exclusively founded on an example of perceptions of the estimations of the incoming factors P. An incoming pattern is depicted by its highlights. These are the qualities of the models for a given issue. The yield of the function could be a persistent worth called regression or could anticipate a class mark of the incoming object termed classifications.

The SVM- Support Vector Machine describes another way to deal with supervised example classification that has been effectively applied to a wide scope of pattern recognition issues. The SVM as supervised ML methodology is alluring in light of the fact that it has an incredibly very much evolved learning hypothesis, factual learning hypothesis. The SVM depends on solid scientific establishments and results in basic yet exceptionally powerful calculations.

The SVM is one of the world's top supervised ML approach, that has accomplished best in class execution on many learning assignments. SVM is effectively utilized for explaining a few NLP assignments for both outside and Indian dialects. Specifically, numerous NLP issues in Malayalam, Telugu and Tamil dialects are understood utilizing SVM device productively than some other administered ML approaches. Each of these dialects just as

Kannada have a place with south Dravidian language class and they take after linguistically and semantically from various perspectives. This is the principle reason behind picking SVM for finding solutions to Kannada NLP assignments.

2. Related Works

Taking a gander at the situation of Indian Languages, we have a few associations inside and outside the nation truly occupied with innovative work in computational semantics and NLP.

As a rule, there are a few methodologies endeavoured for creating morphological analyser and Generator overall [1] [2] [3] [4]. Researchers introduceda two-level morphology approach decade ago, where he tried this formula for Finnish language [5], in two layered representations.Beesleycreated an Arabic limited state transducer for MA in 1996using a Xerox limited state transducer (XFST) [6]. If there should be an occurrence of Indian dialects, the researchers built up a morphological analyser on the basis of limited state automata for Tamil dialect [7]. In the past, small works have been undertaken in computational part of Kannada dialect. The author proposed "System and Process Model" [8] that describes just inflectional morphology. The researchers in IIITH have accomplished few works in regard of Kannada Morphology, however the methodology is paradigm such as the lists based on suffixes.

The framework's execution is limited by volume of the word references & derivational morphology is ignored. Limited state procedure is crucial instrument in the execution of morphology for normal language [9] [10]. The grammatical form tagger for Indian dialects utilizing ML approaches are illustrated in [11] [12] [13]. Authors Leech and Wilson [14] suggested an idea with regard to various levelled label set that is standard rules of EAGLES, which we have adoptedwithout any modifications for the development of our progressive label set for Kannada.

The MA depends on 2data wellsprings: A word reference of legitimate lemmas of a dialect and plenty of rules& regulations for enunciation taking care of. To cover part of formal person, place or thing, the creation happened for a standard based NER as a feature of the annotator system. Since the last decade, NLP researchers have paid more attention to Named Entity Recognition [15], [16] [17]. The analyst in [18] proposed ML approaches for NER, and one more scientist in [19] proposes distinctive blend of ML methods for Hindi dialect.

Fundamental examinations demonstrate thatMA & word references are deficient &blemished and require considerable upgrades for developing a decent Annotation framework. The computational part of Kannada receives very little attention. The creator of "System and Process Model" [20], have accomplished few workswith regard to Kannada Morphology. Our comment framework consists of 5 significant modules such as label set, MA, Named Entity Recognition & word reference. Kannada language is notinvestigated computationally. This situationhas also convinced us to proceed in this direction.The authors Vikram and Shalini built up a model of morphological analyser for Kannada language in view of Finite State Machine [21]. This is only a model dependent on Finite state machines and can all the while fill in as a stemmer, grammatical feature tagger and spell checker. The presented morphological analyser device doesn't deal with compound arrangement morphology and can deal with a limited count of particular things and action words.

The specialists likewise proposed a technique for building solution for Verb Phrase& Noun Phrase understanding in Kannada dialectphrases utilizing CFG [22]. The systemparses the CFG using Recursive Descent Parser which recognizes the syntactic correctness of a given sentence based on the Verb & Noun understanding. Around 200 sample sentences were used to test the structure. As stated in [23] the scientists worked on creating a MA&generators for South Dravidian languages in the year 2010. The MORPH-A system & procedure model for Kannada morphological age/examination was developed and the presentation of the framework is 60 to 70% on general writings [24].

3. Development Of Morphological Generator

Agglutination of Kannada Language

Agglutination is one of themost significant & fundamental feature of Kannada language. It's difficult to set word limits in this language because of its deeply agglutinating feature& morphophonemic variations that occur for the purpose of agglutination. Example,Take the source of action word "ಕೆಲಸಮಾಡಿಕೊಂಡಿದ್ದವನ"(kelasa mAdikoMDiddavana), that is, "the one who was working". The diverse important pieces of this word are as per the following:

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

kelasa mAdu + i + koLLu + MD + u + iru + dd + a + avanu + a

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

Words illustrated above consists of 10 important sets, in that two Verbal Participle (VBP), one root word (Root), two Past Tense Markers (PST), two Auxiliary Verbs (AUXV), one accusative (ACC), one Pronoun (PRON-3SM) and one Relative Participle (RP).

To be precise, there are 3 types of Kannada words: i) naamapada (Nouns) ii) kriyaapada (Verbs) and iii) avyaya (Uninflected words). The Kannada language has three genders: neuter, feminine & masculine. Naamapada and kriyaapada have two kinds of words i.e.singular as well as plural words. No particular distinctive marker is included in the singular words.

Plural marker is as a rule "gaLu", still there are some exemptions such as: Masculine noun like“huDuga” finishing with "an"&few feminine things such as heMgasufinishing with ‘u’contain plural "aru”. The feminine words that end in 'i’ such as “huDugi" or 'e' such as “ate”are pluralized with "yaru". Similarly,when it comes to family relationship wordslike “aNNa”, the plural is frequently "aMdiru". There are a few items thathave irregular plurals, such as, "makkaLu" which is the plural of noun word "magu".

Suffixes of Noun & Characteristics in a variety of cases

In Kannada language, the case structure is similar to the other south Dravidian dialects like Malayalam, Telugu and Tamil. The most common ending for noun words arein a consonant or in a, e, i, u. Different additions are done to the noun stems for demonstratingnumerous connections among noun &sentence’s constituents. Various types of suffixes are usedfor a specific case, in view of kind of nouns&the finishing character. Example, "dative" form trademark addition is chosen by accompanying standards as demonstrated in Table 1., which illustrates the various examples for the noun types along with dative suffixes and dative forms.

Noun Type	Ends With	Dative Suffix	Noun Example	Dative Form
Neuter Noun	(a) ಅ	(kke) ಕ್ಕೆ	(gida) ಗಿಡ	(gidakke) ಗಿಡಕ್ಕೆ
	(e, i, u) ಎ ಇ ಉ	(ge) ಗೆ	(aramane) ಅರಮನೆ	(aramanege) ಅರಮನೆಗೆ
	Consonants	(ige) ಇಗೆ	(janaru) ಜನರು	(janarige) ಜನರಿಗೆ
Neuter Determinative	---	(akke) ಅಕ್ಕೆ	(adu) ಅದು	(adakke) ಅದಕ್ಕೆ
Rational Noun	---	(nige) ನಿಗೆ	(aNNa) ಅಣ್ಣ	(aNNanige) ಅಣ್ಣನಿಗೆ

Table 1: Noun types and Characteristics suffixes for

Dative Case

The Infinitive

Infinitive is indeed a non-limited type of action words that appears in conjunction withother action words (verbs), negative morphemes, helper action words (auxiliary verbs)& few different structures. Kannada has two kinds of infinitives that is "ಓಕ್ಕೆ" (Okke) and "ಅಲ್" (al). Both of these are joined to the action word root to create other word shapes as appeared in Table 2, which illustrates how the infinitive works.

Examples	Illustration
#1	(hogu) ಹೋಗು → go + (Okke) ಓಕ್ಕೆ = (hOgOkke) ಹೋಗೋಕ್ಕೆ → to go
#2	(hogu) ಹೋಗು → go + (al) ಅಲ್ + (illa) ಇಲ್ಲ → negation = (hogalilla) ಹೋಗಲಿಲ್ಲ → did not go

Table 2: Illustration with example for the Infinitive

4. Mag Using Machine Learning Approach

The proposed Morphological Generator is produced using Machine Learning approaches. This area portrays the different endeavours needed for making the suggested rule on the basis of MAG framework. The accompanying datais required for assembling a morphological analyser and generator.The Lexicon, also called as vocabulary, the rundown of stems and affixes along with essential information related to them such as verb stem, noun stem, and so forth. The Morpho tactics is the morpheme model requesting the clarifies which classes of morphemes may follow different classes of morphemes within a single word. For example, standard Kannada plural morphemes comes after the stem connected to noun as opposed to going before it.

The proposed morphological generator model is constructed utilizing supervised ML methodology utilizing the well-known classifications and regressions tool termed SVM. This particular methodology contains the training of

corpus that comprises of sets of incoming objects and the ideal yield. By and large, in morphological examination process, an unpredictable word structure is changed into roots and suffixes. On account of ML method every guidelinesinvolving complicatedrules for spelling are additionally taken care of by the tasks classified. ML moves toward needs just corpora with linguistically data and any hand coded morphological rules are not required. Linguistically or morphological rules are omitted from the commented on corpora. With regards to the proposed strategy, sequence marking is utilized to adjust the equal corpus and morphological examination issue is changed over into classification issue utilizing ML methodology.

The exhibition of the morphological analyser incredibly relies upon the adjusted morphological corpora that ought to be enormous, delegate & acceptable quality. Series of stages engaged with corpus improvement is demonstrated in Figure 2. The Support Vector Machine bolster only the Roman symbol code, yet Dravidian dialectslike Kannada doesn't support this format of code and bolster just Unicode symbol. Mapping documents were made and utilized to outline Unicode to Roman and the other way around. The Romanized adjusted input-yield information corpus, consisting of most regularly utilized action words, chose from all action word standards was made physically.

This segmentation's step includes four unique stages: grapheme division, parting syllable, representation and categorization of Vowel and Consonant. Every Romanized word in the corpora is divided into Kannada grapheme in the main stage. In the next stage, these graphemes are divided into syllables of vowels & consonants. The vowels &consonants markers – "V" &- "C" are applied to the brokenvowel &consonant syllable separately in the next stage. Character "*" is utilized for showing the morpheme limits of the yield data. There are existing methodologies which we utilized to contrast our frameworks. The primary methodology is called corpus based methodology where a huge estimated all around created corpus is needed for training, utilizing the ML calculations.

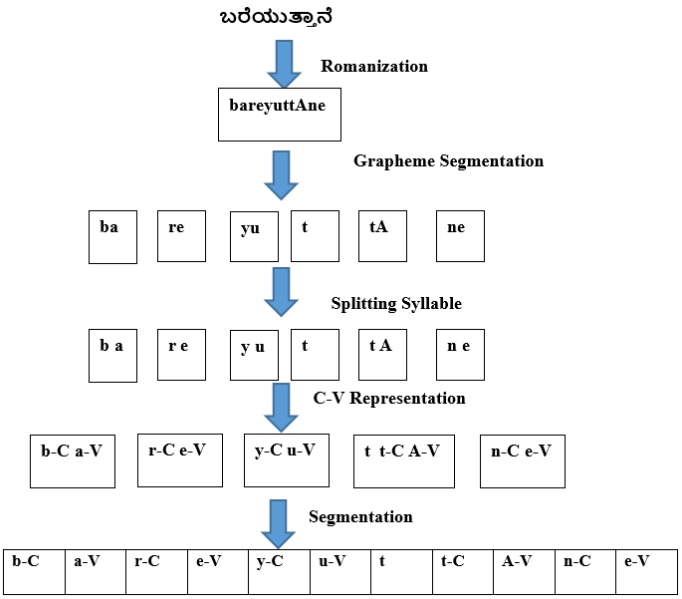


Fig. 2: Numerous steps involved in corpus development

The presentation of the framework will rely upon the size as well as element of the corpus. The challenge is that constructing a corpus is a time-consuming process. Then again, rule dependent methodologies are based with respect to a lot of decides and word reference that contains roots and morphemes. In rule dependent methodologies each standard depends with respect to the past principle. So in the event that one principle fizzles, it will influence the various standards that tail it. At the point when a word is given as a contribution to the morphological analyser and in the event that the relating morphemes are absent in the word reference, at that point the standard based framework comes up short.

In the proposed framework, the framework takes data from pictures/message and create Unicode for that Kannada picture. Morphological analyser issue is reclassified as characterization issue and understood utilizing ML system. This is a corpus dependent technique in which Support Vector Machine calculations are used training and testing. The performance of the framework is additionally contrasted and different frameworks created utilizing a similar corpus and results demonstrated that SVM based methodology beats the other. The stepwise representation of development of Morphological generator is shown in Fig.3 In this figure, after Romanization and segmentation, the next stage is Alignment and Mapping which uses the succession marking approaches the

sectioned input- yield words which were adjusted vertically and subsequently as fragments with space between them.

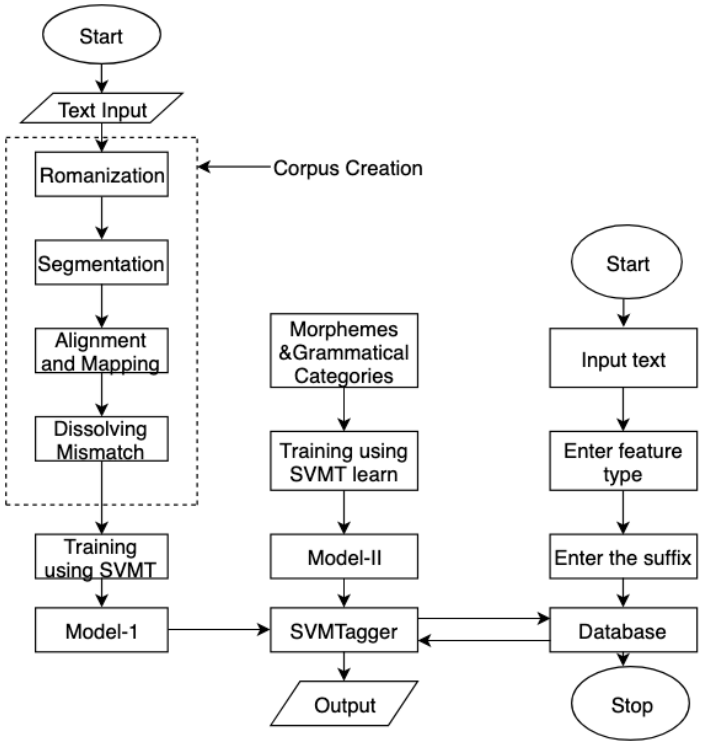


Figure 3: Block Diagram showing Morphological Generator development stages

While we map inputs and yield information terms, problem of confounding may occur either because the information units are larger or smaller than the yield units. Pre-handled parallel system consists of grouping of information characters and their corresponding yield names. Parallel corpus comprises more than 300,000 words which were prepared utilizing SVMT learning device & MA model named Model-I was created.

Model-I was used for breaking down & recognize various morphemes associated with input testing words given. As a result,a new model known as model-II was created for assigning syntactic classes to every word morpheme and this model was created by grouping morphemes and their linguistic categories. The following is the working rule:The prepared Model-I receives the testing input terms. Every mark to the information section is predicted by the prepared Model-I. The divided morphemes are then given to the prepared model-II in the next stage. For the given input terms, itforecasts syntactic classifications for the portioned morphemes.

The Morphological Analyser is created as computer application, where, at the point when client clicks Analyser button, the Morphological Analyser module peruses the recognized word and divide the given term to part morphemes & appointing correct morpho-syntactic information. Consider Example of giving input word as "Anegalu", The yield of Morphological Analyser module is given as: Stem/Root: "Ane", Number: Plural, Case: Nominative, Gender: neuter. The working algorithm is devised in the form of steps given below:

Step1: Models are provided with training sets with comparing bunch vector and do the classification of test sets utilizing a SVM classifier.

Step2: Construct models

Step3: Vectorised instruction that binarizes Group in which 1 is the present class and 0 is all different classes

Step4: Perform the classification of test cases.

The another module of the built application, Morphological Generator works as follows:

At the point when client clicks Generator button, the Morphological Generator gets base expression of a thing, morphological highlights like case, gender, and number as contribution from the client, and afterward, it gives the genuine word as an output. Consider an example, input: (book), "pustaka"[base], [Case] Nominative, [Number] Plural, [Gender] Neuter, the yield is: "pustakagaLu" [w]. The corresponding algorithmic representation is given below:

Reason: To create a word for given base and morphological component.

Info: morphological element for a word, for example, Gender, Case, Base and Number

Yield: Word w.

Steps:

• Extract gender, case, number from classification code

• Derive stem from base utilizing code for modifier

• If (discovered) at that point return (suffix that is comparing to morphological component)

• Do concatenation of stem with postfix to shape w.

5. Experimentation And Result Analysis

The development of MAG is a difficult errand for a wide range of word structures. The created rule-based MAG is fit for dissecting and producing a rundown of more than thirty thousand nouns, around 4,000 action words and a moderately littler list of adjectives. Theability of the produced MAG to generate and analyse causative, transitive, and tense structures apart from auxiliaries, disconnected developments and verbal items in its distinguishing feature.

Tool, Morphological analyser and generator for nouns of Kannada is made using the best strategies of ML. The examination and generation issues are re-framed as issues of classifications. The system relies upon progression stamping and planning by piece strategy which gets the nonlinear associations of the morphological features from getting ready data tests in predominant and less troublesome way. Use is anticipated to meld progressively linguistic information of Kannada dialect with incredible conceptual features, that deals with morphological problem even more satisfactorily.

Constructed MAG is prepared for separating and delivering thousands of nouns. Similarly, the exactness of the rendered MAG could be improved by comparing it to consistently increasing number of different types of word comparisons. We started the research work as an application development projectby setting up the runtime requirements for working in MATLAB 2016b in windows system. This made it easy for us to carry out the calculations &create the required user interfaces. Kannada applicationis used everywhere we need to sort terms in kannada dialect.

Input information collections chosen for evaluating constructed application assumes the critical job in acquiring the proficiency of apps. Input data in the form of text& pictures is catch archives, obtained fromorganizers or informationstructure on the consoles. The system advancement is used to conduct the emphasis-based evaluation.Correlations of the yield delivered by machines with anticipated yield are used to evaluate the analysis tool. Structures that are not needed as well as words that are incorrectly pronounced are removed&reconnection is rendered as per the to thestem classificationpractises. Testing of the application is important to understand the validation and verification of the works done. The input data sets opted for validating the system developed makes major role in obtaining the application's efficiency. The inputs are obtained from various means, such as by capturing the kannada docs, by browsing the kannada docs locally, by typing texts using keyboards. The Table 3, shows the results of validation for few test cases for developed MAG using SVM. The results illustrate that the expected output is predicted correctly.

Test Cases	Feature value	Expected output	Predicted output
TC#1	Root=Ane Number=Plural case=Locative Gender=Neuter	AnegaLalli	AnegaLalli
TC#2	Root=sahana Number=Singular case=Genitive Gender=Female	sahanaLa	sahanaLa
TC#3	Root=rAma Number=Singular case=Accusative Gender=Male	rAmanannu	rAmanannu
TC#4	Root=aNitha Number=Singular case=Dative Gender=Female	aNithaLige	aNithaLige

Table 3: Validation results of MAG

The diagrams from Figure 4 to Figure 10 shows the snapshots of GUIs&usage of MA& Generator function.

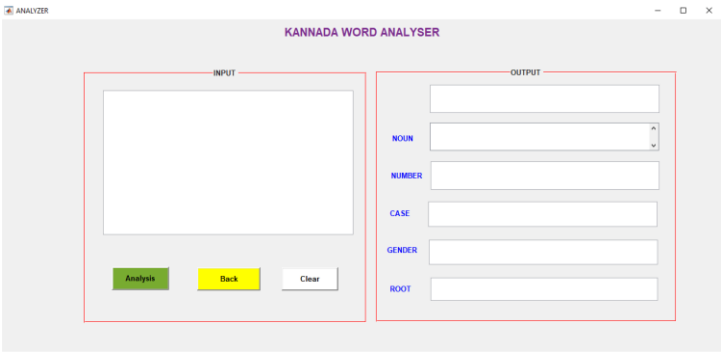


Figure 4: Screenshot-1 usage of MA& Generator tool

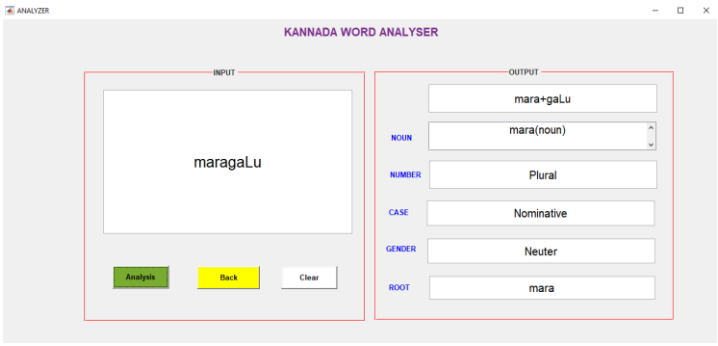


Fig. 5: Screenshot-2 Generator tool &Morphological Analyserusage

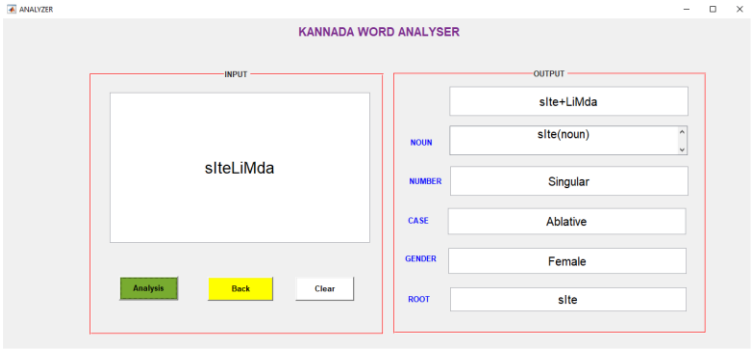


Figure 6: 3rd screenshot of usage of MA&Generator method

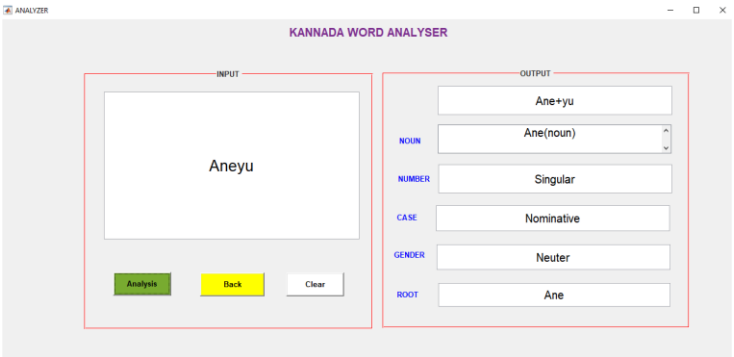


Figure 7: Screenshot-4 of MA& Generator method

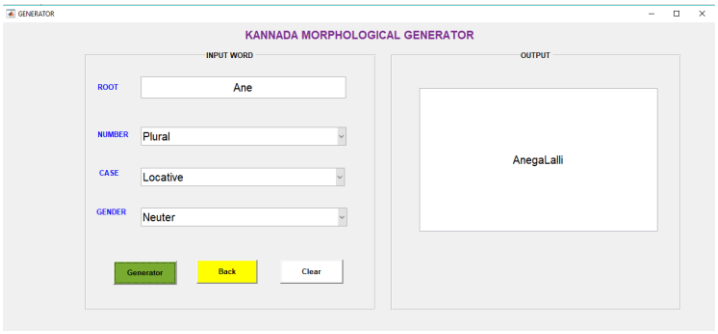


Figure 8: Screenshot-5 of MA& Generator technique

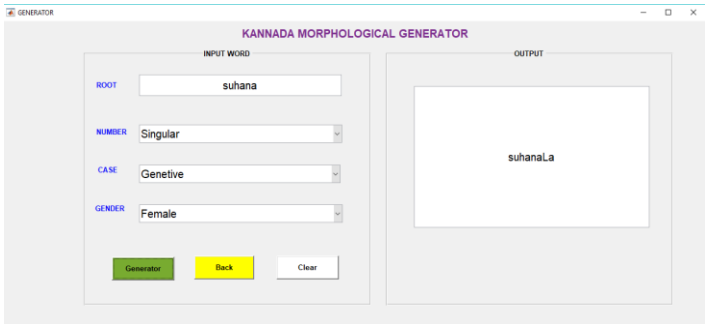
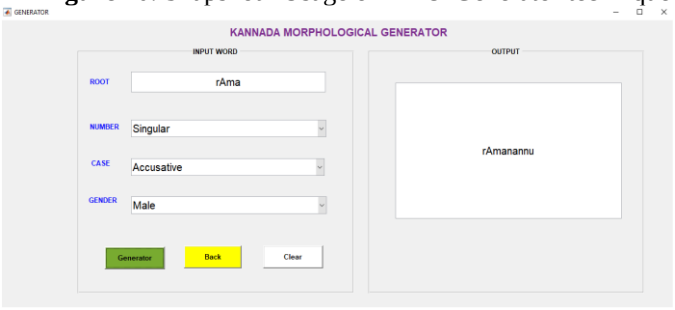


Figure 9: Screenshot6- Generator tool &Morphological Analyserusage

Figure 10: Snapshot7 Usage of MA& Generator technique



Multi SVM algorithm classified the Kannada noun wordssuccessfully. The major number of misclassifications is reduced for analyser as well the generator. Accuracy is achieved to the maximum possible rate, making use of the SVM strategies, by using minimal rules for both training the input sets as well getting the outputs.

The behaviours of the eight different input data sets which were used in our previous publication is tabulated in Table-4, Which includes the number of words in the input data groups, quantity of unanalysed terms andwhich are inserted in lexicon. The graphs illustrated in Fig.11 and Fig.12, gives the analysis of words which are unanalysed and inserted in to lexicons, and the number of words which are inflectional versus non inflectional.

Input Data Sets	No. of words in the input data sets	No. of words unanalysed	No. of words Inflectional	No. of words Non-inflectional	No. of Non-kannada words (Foreign)	No. of words unanalysed inserted in to lexicon
#1	1500	26	22	4	6	20
#2	1100	34	28	6	8	26
#3	350	16	13	3	5	11
#4	400	28	22	6	8	20
#5	700	32	27	5	9	23
#6	2600	92	69	23	22	70
#7	1850	63	48	15	16	47
#8	1450	52	42	10	14	38

Table 4. Performance of Insertion of lexicons of unanalysed Words into MAG

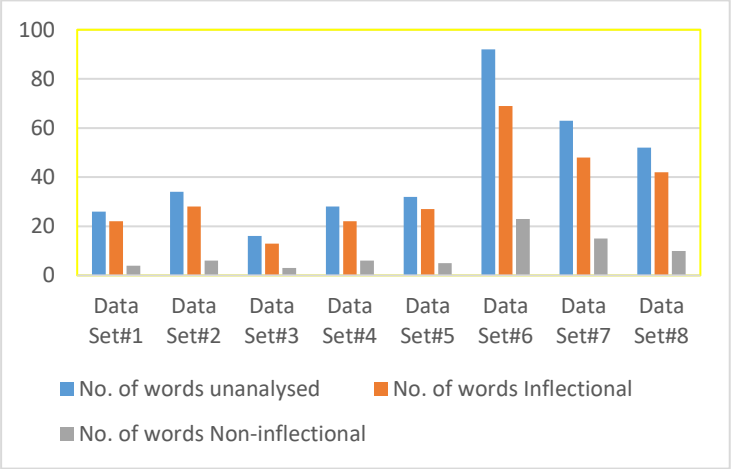


Figure 11: Graphical analysis of No. words unanalysed, inflectional and Non-inflectional

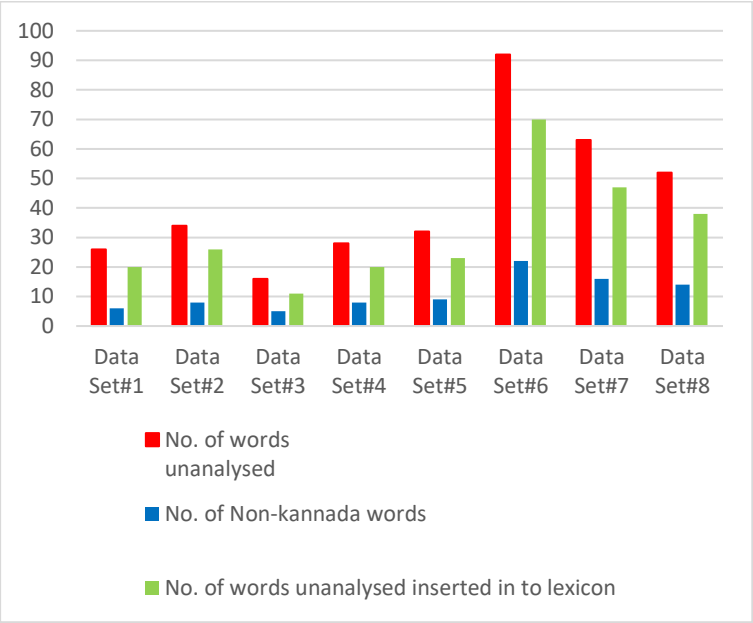


Fig. 12: Graphical analysis of No. of words unanalysed, Non-kannada and unanalysed inserted in to lexicon

The evaluation of inclusion of lexical subtleties of unanalysed words into Kannada monolingual lexicon unit is determined as far as number of unanalysed words from MAG are embedded into lexicon. The subtleties of unanalysable words like inflectional, non-inflectional and remote words are appeared in Table-4. After inclusion of unanalysed words into lexicon, execution of MAG is improved. It is seen from the Tabulation and analysis that, performance of MAG has been expanded well, after addition of lexical subtleties of unanalysed Kannada words into lexicons.

The F-measure, accuracy, precision& recall are all measures as a part of the MAG's performance assessment using the below formulae.

*Accuracy for Normalizer = NWt / NWi * 100*

*Accuracy for Morphological Stemmer = NWs / NWu *100*

*Precision = TPR/ (TPR + FPR) *100*

*Recall = TPR / (TPR +FNR) * 100*

*F_measure = 2 * Precision * Recall / (Precision + Recall) * 100*

Where:

NWi - Number of words input

NWs - Number of stemmed words correctly

NWu - Number of words unique inflectional

NWt - Number of tokenized words correctly

TPR - True Positive Rate (Number of words analysed correctly)

FNR - False Negative Rate (Number of words unanalysed)

FPR - False Positive Rate (Number of words analysed incorrectly)

Tabulation of results for the 8 input sets, which computes F-measure, recall and accuracy rates based on the quantity of words rightly analysed, wrongly analysed and which are unanalysed, given in the Table 5.

Input Data Sets	No. of words in the input data sets	No. of words rightly analysed (TPR)	No. of words wrongly analysed (FPR)	No. of words not analysed (FNR)	Precision percentage	Recall percentage	F_measure Percentage
#1	1500	1396	98	6	93.44	99.57	96.41
#2	1100	1006	82	12	92.46	98.82	95.53
#3	350	328	17	5	95.07	98.50	96.75
#4	400	350	42	8	89.28	97.77	93.33
#5	700	663	27	10	96.09	98.51	97.28
#6	2600	2541	45	14	98.26	99.45	98.85
#7	1850	1812	26	12	98.59	99.34	98.96
#8	1450	1355	86	9	94.03	99.34	96.61

Table 5: Performance evaluation of MAG after Insertion of Unanalysed Words into the Lexicon

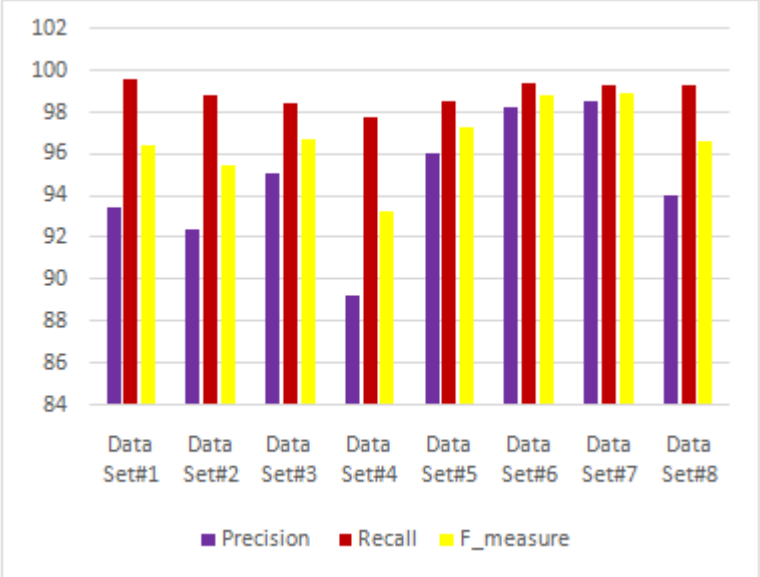


Fig. 13: Graphical analysis of MAG performance, after Insertion of Unanalysed Words into the Lexicon

Figure 13 demonstrates the graphical representation of tabulated outcomes. Precision posted the highest value (98.59%) for input data set#7 &lower value (89.28%) for input data set#4. Maximum of (99.57%) for input data set#1 & least value (97.77%) for input data set#4 is reported by Recall. The F_measure recorded amaximumof (98.96%) for input data set#7, least value of (93.33%) for input data set#4.

6. Conclusions

The Morphological Analyser and Generator tool is developed is having capacity of analysing and generating a record of thousands of kannada texts. The maximum required accuracy has been reached by using ML approaches. The results are achieved by using less number of rules for training input data sets as well to obtain the results. The developed module is reliable enough to predict many of the kannada texts from the input set.The precision of MAG produced could also be improved by testing it against as many different types ofword lexicons as possible..

References

1. John Chen et al. “Towards automatic generation of natural language generation system,". Proc. of the 18th International Conference on Computational Linguistics, New York City, USA, 2000.

2. Goldsmith and Gaussier, “Unsupervised learning of derivational morphology from inflection lexicons”. Proc. of ACL Workshop proceedings, pp. 24–30, 1999.
3. Goldsmith, “Unsupervised learning of the morphology of a natural language". Computational Linguistics,pp. 153–193, 2001.
4. Creutz M, “Unsupervised segmentation of words using prior distributions of morph length and frequency”. Proc. of the Association for Computations Languages (ACL03), pp. 280–287, Sapporo, Japan, 2003.
5. K. Koskenniemi, “Two-level morphology: A general computational model for word form Recognition and production”. Master’s thesis, Department of Genera Linguistics, Helsinki University, 1983.
6. Kenneth R Beesley, “Arabic morphology using finite state operations” ph.D Thesis, 1983.
7. Vijay and Shobha. Tamil morphological analyzer. http://nrcfosshelpline.in/smedia/images /downloads/Tamil Tagset-opensource.odt
8. Asahara, Masayuki and Matsumoto. “Japanese Named Entity Extraction with Redundant Morphological Analysis”. Proc. of Human Language Technology conference – North American chapter of the Association for Computational Linguistics, 2003.
9. E Roche and Y Schabes, “Introduction finite state language processing”. MIT Press, 1997.
10. Hopcroft J.E and J D Ullaman., “Introduction to automata theory languages and Computation”. Addition Wesley, 1979.
11. Sivaji Bandyopadhyay, Asif Ekbal, and Debasish Halder, “Hmm based pos tagger and rule-based chunker for Bengali”. Proc. of NLPAI Machine Learning Contest 2006.
12. Sandipan Dandapat and Sudeshna Sarkar, “Part of speech tagging for Bengali with Hidden Markova model”. Proc of NLPAI Machine Learning Contest 2006.
13. Aniket Dalal, Kumar Nagaraj, Uma Sawant, and Sandeep Shelke, “Hindi part-of speech tagging and chunking: A maximum entropy approach”. Proc. of NLPAI Machine Learning Contest, IIIT, Hyderabad, India, 2006
14. Leech G and Wilson, “A. Recommendations for morph syntactic annotation of corpora”. EAGLES Technical report, 1996.
15. YUNGWEI DING HSINHSI CHEN and SHIHCHUNG TSAI. “Named entity extraction for information retrieval”. Proc. of HLT-NAACL, pp 8-15, 2003.
16. Grishman. “The nyu system for muc-6”. Proc. of Sixth Message Understanding Conference (MUC-6), pp 167–195, Fairfax, Virginia. 1995.
17. Kashif Riaz. “Named Entities Workshop”. Proc. of Association for Computational Linguistics ACL, Uppsala, Sweden., pp. 126–135, 16 July 2010.
18. Ekbal and S. Bandyopadhyay. “Named entity recognition in Bengali: A Conditional random field”. Proc. of ICON, pp 123–128, 2008.
19. Michael Fleischman. “Automated sub categorization of named entities”. Proc. of Conference of the European Chapter of Association for Computational Linguistic, pp 25–30, 2001.
20. Mukund Sangalikar, Shilpi Srivatsava and D.C. Kothari. “Named entity recognition System for Hindi language”. International journal of Computational Linguistics vol. (2), pp 10–23, 2011.
21. Kavi Narayana Murthy. A network and process model for morphological analysis/generation. In Proc. International Conference on South Asian Languages, Punjabi University, Patiala, India,9-11 January, 1999.
22. T. N. Vikram & Shalini R Urs, “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, 2007.
23. B.M. Sagar, Shobha G and Ramakanth Kumar P,“Solving the Noun Phrase and Verb Phrase Agreement in Kannada Sentences”, International Journal of Computer Theory and Engineering, Vol. 1, No. 3, 1793-8201, August 2009.
24. www.languageinindia.com/may2011/v11i5may2011.pdf.
25. K. Narayana Murthy, “A Kannada morphological analyzer and generator using Network and Process Model”, Resource Centre for Indian Language Technology Solutions–Telugu University of Hyderabad, http://www.tdil.mit.gov.in/Telugu-UOHJuly03.pdf