MORPHEME BASED PARTS OF SPEECH TAGGER FOR KANNADA LANGUAGE

[1]M. C. PADMA, [2]R. J. PRATHIBHA

[1]P. E. S. College of Engineering, Mandya, Karnataka, India
[2]S. J. College of Engineering, Mysore, Karnataka, India
E-mail: [1]padmapes@gmail.com, [2]rjprathibha@gmail.com

Abstract- Parts of speech tagging is the process of assigning appropriate parts of speech tags to the words in a given text. The critical or crucial information needed for tagging a word come from its internal structure rather from its neighboring words. The internal structure of a word comprises of its morphological features and grammatical information. This paper presents a morpheme based parts of speech tagger for Kannada language. This proposed work uses hierarchical tag set for assigning tags. The system is tested on some Kannada words taken from EMILLE corpus. Experimental result shows that the performance of the proposed system is above 90%.

Index Terms- Hierarchical tag set, morphological analyzer, natural language processing, paradigms, parts of speech.

I. INTRODUCTION

Parts of speech tagger or annotator is a tool which assigns the appropriate syntactic categories to the words in a given text. Parts of Speech (PoS) tagger plays an important role in most of the Natural Language Processing (NLP) applications like information retrieval system, machine translation system, word sense disambiguation system, etc.,. In general, supervised and unsupervised approaches are used for PoS tagging. The supervised technique requires annotated data set to train the system but unsupervised PoS tagging method does not require previously annotated data set. PoS tagging methods once again fall under three categories, viz., rule based or linguistic based, stochastic or data-driven based and hybrid. In rule based method, set of hand written linguistic rules are framed based on the morphological and contextual information. In stochastic method, frequency based information is derived from the previously trained data. Hybrid tagger combines the features of both rule based and stochastic based approaches. Kannada is one of the Dravidian languages spoken primarily in South India. Kannada is a classical and administrative language in Karnataka. Kannada is an inflectional, derivational, morphologically rich and relatively free word order natural language. Normally, the main verb is in terminating position and the remaining words of all other lexical categories and sub-categories can occur in any position in the sentence. During the generation of inflectional words, the morpheme components like prefix, derivational suffixes and/or inflectional suffixes are attached to a root. Generally, the critical or crucial information required for correct tagging a word comes from its internal structure rather from its context in the given sentence. In most of the cases, information required for disambiguating tags comes from internal structure of the word, not from its neighboring words. Hence, morphological analysis is very essential in determining the PoS category of a word. A tag set is generally chosen based on the language application for which PoS tags are used. In this paper, we propose a morpheme based PoS tagger for Kannada Language using Board of Indian Standards (BIS) Dravidian tag set.

II. LITERATURE SURVEY

In general, several methodologies are used in the development of PoS taggers for Indian and non-Indian languages. Brill tagger designed for English is a rule based tagger, which uses hand written linguistic rules to assign tags to the given words [1]. Hindi PoS tagger uses a set of linguistic transformation rules to assign appropriate tags [2]. PoS tagger for Tamil language is designed, using morphological features of Tamil words and obtained F-measure of 96% [3]. A hybrid PoS tagger for Malayalam is proposed, using Conditional Random Field (CRF), Support Vector Machine (SVM) and rule based approaches [4]. A morphology based automatic PoS tagger is designed for Telugu by extracting the morphological features of Telugu words [5]. Several PoS taggers are developed for Kannada language, using machine learning approaches like SVM, CRF, HMM, maximum entropy and rule based etc.,. A maximum entropy based PoS tagger is designed by taking 51267 words from the EMILEE corpus as training data set. The tag set contains 25 tags. The system is tested on 2892 words downloaded from Kannada website and obtained accuracy is 81.6% [6]. The second order HMM and CRF PoS tagger for Kannada is proposed and obtained the accuracy of 79.9% and 84.58% respectively [7]. A CRF based PoS tagger is designed by collecting 1000 words from on-line Kannada news paper to train the system. The training data set is tagged manually using tag set which contains 45 tags. The accuracy, obtained by this system is 99.49% [8]. A rule based PoS tagger proposed by using morphological features and hierarchical tag set [9]. In this work, the morphological system is designed using finite state transducer and obtained accuracy of 90% for nouns and 85% for verbs. Most of the existing PoS taggers for Kannada language are generally designed using machine learning approaches like HMM, CRF, maximum entropy and SVM. These algorithms require an extensive training data set to train the system. The performance of such taggers directly depends on the size of the training data set. The standard pre-tagged Kannada corpus is not available publicly. Hence, the training data set must be tagged and verified manually. However, only the words that are already trained will be identified, recognized and tagged correctly using machine learning approaches. Hence, the performance of Kannada PoS taggers is directly proportional to the size and content of the training data set. Since Kannada is an inflectional, derivational and morphologically rich language, all declension forms of inflectional and derivational nouns and verbs cannot be included in training data set. However, in case of morphologically rich and relatively a free word order language like Kannada, the critical or crucial information required for correct tagging a word comes from its internal structure rather from its neighboring words in the given sentence. Hence, this paper proposes a morpheme based PoS tagger for Kannada language by extracting morphological features and grammatical information from the input words. This proposed work uses the BIS Dravidian tag set for assigning tags. The BIS Dravidian tag set is a hierarchical tag set containing 26 tags.

III. PROPOSED WORK

A. Architecture of the Proposed Model

The architecture of the proposed system is shown in Figure 1. The system consists of two modules and three tables. The two modules are i) text preprocessor and ii) derivational and inflectional morphological analyzers. The three databases constructed specifically for the proposed system are i) encoded suffix table, ii) look-up table and iii) Kannada monolingual lexicon. The details of the content of the tables are explained below.

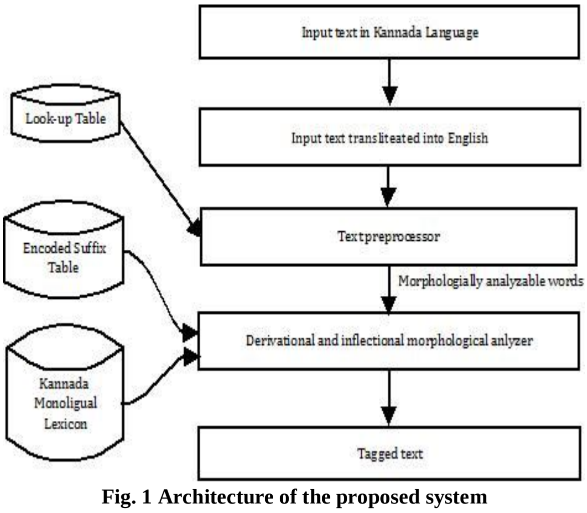

Input text in Kannada Language
Look-up Table
Input text transliteated into English
Encoded Suffix Table
Textpreprocessor
Morphologially analyzable words
Derivational and inflectional morphological anlyzer
Kannada Monoligual Lexicon
Tagged text

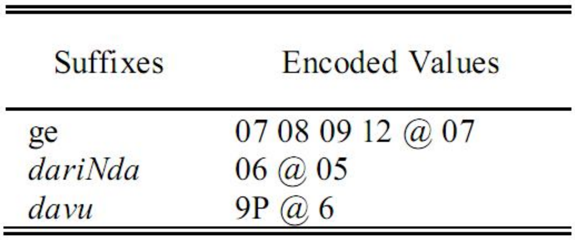
Fig. 1 Architecture of the proposed system

B. Databases Created for the Proposed Model

1. Creation of Encoded-Suffix Table using Paradigm Based Approach.

All available Kannada inflectional and derivational suffixes of nouns and verbs are classified into set of paradigm classes using paradigm-based approach. In this proposed work, an encoded-suffix table is constructed which contains list of suffixes and their lexical features in encoded form. The lexical features are: a set of paradigm-class numbers to which the suffix belongs and position of the suffix in the noun and verb paradigm classes [10-11]. For example, the suffix ‘ge’ is appeared in noun paradigm-classes 07 08 09 and 12 at 07th position, hence the encoded value for the suffix ‘ge’ is 07 08 09 12 @ 07 as given in Table 1. The position of suffix is used to derive the grammatical features like case and number for noun; and person, number and gender for finite verb. Few entries of the encoded-suffix table are shown in Table 1.

TABLE I ENCODED-SUFFIX TABLE

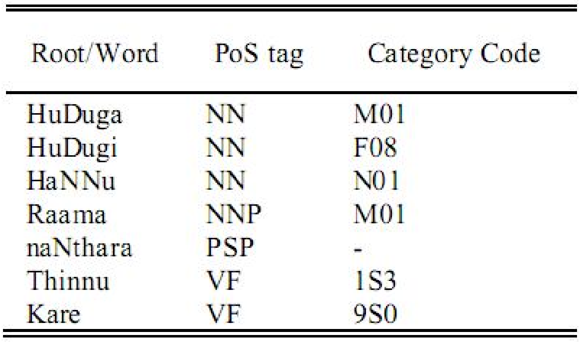


Suffixes	Encoded Values
ge	07 08 09 12 @ 07
dariNda	06 @ 05
davu	9P @ 6

2. Creation of Kannada Monolingual Lexicon using Rule Based Approach.

Kannada monolingual lexicon is a dictionary which contains the Kannada root or base form of word, its PoS tag and lexical details in English transliterated form. The lexical details for noun are gender and paradigm-class [10-11], for finite verb, paradigm-class and modifier-code. For example, huDugi belongs to feminine gender and 08 paradigm class; hence its category code is “F08”. For verbs, the first two characters of the category-code represent the paradigm-class to which it belongs and the last character is a modifier-number. The modifier-number indicates the number of characters to be removed from the verb-root to get the stem for past tense verb inflections. For example, the verb root ‘thinnu’ belongs to the paradigm-class “1S3”. Here 3 is the modifier-number. This number represents the last three characters of the root to be stripped off. Hence the remaining characters ‘thi’ is the stem for past tense, from which verb forms for the past tense can be inflected. The past tense forms of the verb-root ‘thinnu’ are ‘thiNdanu’, ‘thiNdaLu’, ‘thiNdithu’, and so on. Hence, there is no need to store all inflectional forms of nouns and verbs in the lexicon. All inflectional forms of both nouns and finite verbs are generated with the help of encoded suffix table and lexicon. This reduces the space complexity extensively. Rule based approach is used to create the Kannada monolingual lexicon. The root words are randomly selected from a well known Kannada dictionary called Kannada Rathna Kosha [12]. Currently the lexicon consists of 3500 root words with their lexical features. Some of the entries of Kannada monolingual lexicon are given in Table 2. The notations that are used in Kannada monolingual lexicon are; M - Masculine, F – Feminine, N - Neuter and S - Past tense.

TABLE 2.
KANNADA MONOLINGUAL LEXICON

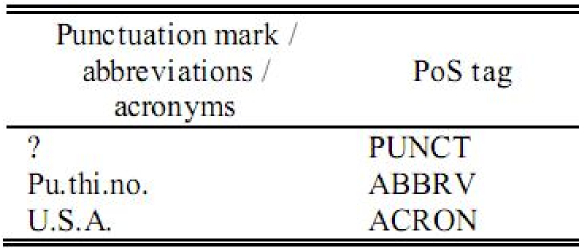


Root/Word	PoS tag	Category Code
HuDuga	NN	M01
HuDugi	NN	F08
HaNNu	NN	N01
Raama	NNP	M01
naNthara	PSP	-
Thinnu	VF	1S3
Kare	VF	9S0

3. Creation of Look-up Table

The look-up table contains punctuation marks, abbreviations and acronyms of Kannada language with their respective PoS tags PUNCT, ABBRV and ACRON respectively. These details are manually entered and stored in look-up table. Some of the entries of look-up table are shown in Table 3.

TABLE 3. LOOK-UP TABLE



Punctuation mark / abbreviations / acronyms	PoS tag
?	PUNCT
Pu.thi.no.	ABBRV
U.S.A.	ACRON

4. PoS Tag Set

In order to assign an appropriate tag to a token, it is necessary to have a tag set. In the proposed work, the BIS Dravidian tag set is used. It is a hierarchical tag set. The list of hierarchical tags with their subtypes, tag label and examples is shown in table 4.

C. Methodology used

The proposed system gets the input text in Kannada language. From the computational perspective, the input text is transliterated into English form using transliteration tool [13]. The preprocessing module tokenizes the given input text into set of tokens using Indic tokenizer [13]. The Indic tokenizer is a special tokenizer which is specifically designed for tokenizing the text in natural language processing applications. The preprocessing module also handles the tokens that have no morphemes, like punctuation marks, symbols, acronyms, abbreviations etc, by directly searching them in look-up table.

The token that contains morphological features is given to derivational and inflectional morphological analyzer module. In this module, if the word is in its base (no affixes) form, then it is directly searched in Kannada monolingual lexicon. If the word is found in the lexicon, then its lexicon tag is assigned as the PoS tag. The morphological analyzer module searches for the existence of derivational and/or inflectional affixes in the given input word using affix-stripping approach and then split the given inflectional/derivational word into prefix, stem and suffix. Initially, this module searches for presence of prefix and/or suffix in the given input word using prefix-list, derivational suffix-list and encoded-suffix table. If prefix and/or suffix are found, then it extracts the stem from input word by stripping off the affixes, and then returns paradigm-classes and position of the suffix from encoded-suffix table. The extracted stem need not be linguistically meaningful. If no affixes are present in the input word then it is considered as base or root or indeclinable word. The position of the suffix is used to derive case and number if the suffix belongs to noun, otherwise person, number and gender (PNG) information is derived if the suffix belongs to verb. This module also tests the correctness of the inflectional formation of input inflected word by considering the paradigm-class of stem and suffix. Initially this module searches for the stem in the Kannada monolingual lexicon. If it is found, then it gets the corresponding lexical category-code and PoS tag. If the input word is an inflectional word and its paradigm-class, gender and/or modifier-number are extracted from the lexical category-code. If the input paradigm-class and the extracted paradigm-class are same, then the input word is a valid inflectional word. If the input word belongs to noun category then case and number of the input word are derived from the position of the suffix and PoS details of the input word - "NOUN, prefix, stem, suffix, gender, case and number" are displayed, otherwise person, number, gender (PNG) and tense of the input word is derived from the position and paradigm-class of the suffix. The details of finite verb - "VERB, word, stem, suffix, person, number, gender and tense" are displayed.

Some of the prefixes in Kannada are: "pra, "paraa", "Apa" ,"sam", "Ava", "nis", "nir", "dus" ,"Abhi", "prathi", "pari", "upa" ,"A","vi" ,"Adhi" ,"Athi" ,"uth" ,"su" ,"dur " ,"Anu" ,"Athi" ,"ni" ,"ku". Some of the Kannada derivational suffixes are: "gaara" ,"yaaLu" ,"vaNtha", "vaNthe", "daara", "kaara", "koora", "thana", "iikaraNa", "aathiitha", "vaada","para","shaahi","gattale".

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The proposed system is tested and experimental results are obtained. The best sample input text containing all kinds of words like inflectional words, numbers, punctuation marks, acronym, abbreviation etc., which is used to test the proposed work is given below in Kannada font and English transliterated form.

ರಾಜ್ಯದ ೨೫೮ 'ಸರಕಾರಿ' ಕೇಂದ್ರಗಳಲ್ಲಿ ಪರೀಕ್ಷೆಗಳ ತರಬೇತಿಯನ್ನು ಡಾ. ವಿದ್ಯಾಭೂಷಣ (ಸಿ.ಇ.ಒ.) ಅವರು ನೀಡುತ್ತಾರೆ.

raajyada 258 “sarakaari” keeNdragaLalli pariikShegaLa tarabeethiyannu Dr. vidyabhushana (C.E.O.) Avaru niiDuthtaare.

D. Experimental Results

In the preprocessor module, the Indic tokenizer splits the given input text into set of sixteen tokens. Out of these twelve tokens, five tokens that do not have morphological features are assigned relevant PoS tags using look-up table and one token (258) is assigned tag as NUMB using regular expression.

Output of preprocessor module is shown below.

I) Set of words in the given input text, that do not have morphemes and are given tags by the preprocessor module are given below.

258 : NUMB
“, “ , (,) . : PUNCT
Dr. : ABBRV

II) Set of words in the given input text that are given to inflectional and derivational morphological analyzer are as follows.

raajyada sarakaari keeNdragaLalli pariikShegaLa tarabeethiyannu vidyabhushan Avaru niiDuthtaare.

The output of derivational and morphological analyzer for assigning PoS tags to the input words is shown below.

1. raajyada <raajya: NN-COM-N-SL-ABL>
2. sarakaari <sarakaari: NN-COM-N-SL>
3. keeNdragaLalli <keendra: NN-COM-N-PL-LOC>
4. pariikShegaLa: <pariikShe: NN-COM-N-PL>
5. tarabeethiyannu: <tarabeethi: NN-COM-N-SL-ACC>
6. vidyaabhuushana <NNP-M-SL>
7. Avaru: <PRP-PL-NOMl>
8. niiDuththaare: <niiDu: VF-FP-P-M-PR>

E. Performance Evaluation of the Proposed Model

To test the performance of the proposed system, four different data sets are created from Enabling Minority Language Engineering (EMILEE) corpus. The EMILEE corpus is a collaborative work of researchers at Lancaster University, United Kingdom and Central Institute of Indian Languages (CIIL), Mysore, India. These data sets contain different types of words like nominal, adjectival, pronominal, verbal inflectional words and derivational words. The proposed system is evaluated by considering the parameters precision, recall and F-measure using the following equations (1), (2) and (3).

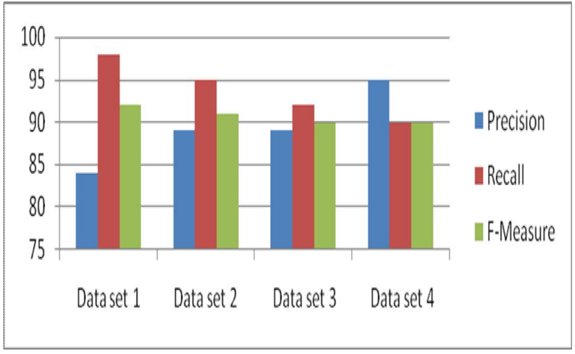
$$Precision = \frac{Tp}{Tp + Fp} * 100 \qquad (1)$$

$$Recall = \frac{Tp}{Tp + Fn} * 100 \qquad (2)$$

$$F - measure = \frac{2 * precesion * recall}{recision + recall} * 100 \qquad (3)$$

Where
Tp – True positive: Number of words correctly tagged
Fp – False positive: Number of words wrongly tagged
Fn – False negative: Number of words untagged

The confusion matrix containing the result analysis of the proposed system on four different dataset is given in Table 5. The graph plotted for the obtained result is shown in Figure 2.



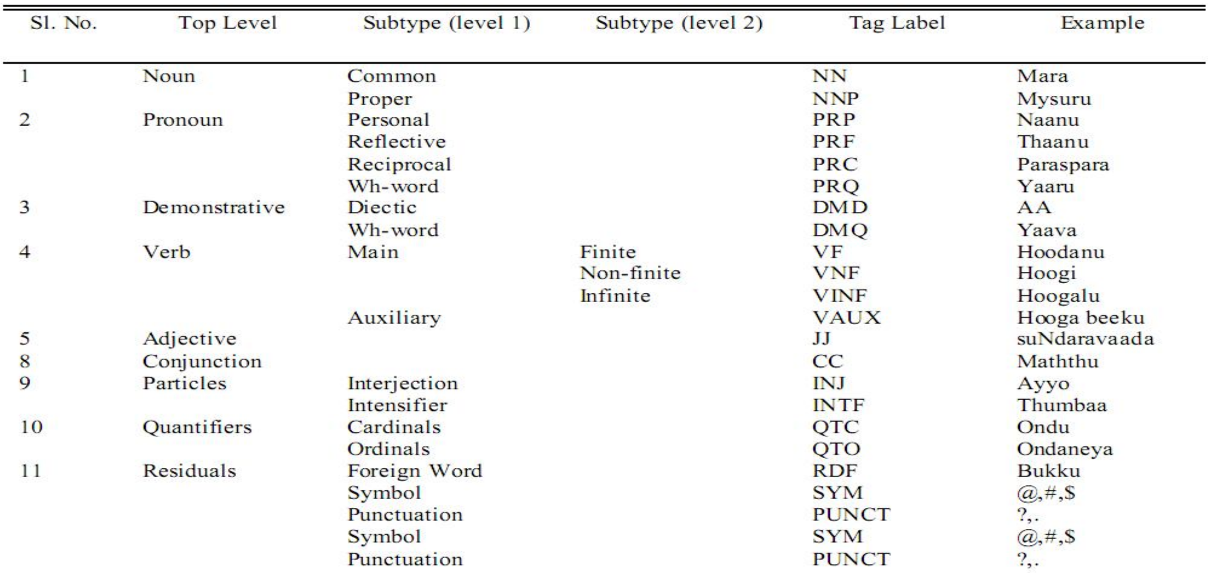
F. Discussion

It is observed from the confusion matrix that, F-measure value computed by the proposed system is directly proportional to the size of the data in the input data sets. As the number of words increases in the input data set, the F-measure value computed by the proposed system increases. However, this is not true always because the performance of the proposed system depends on the corpus that is used for testing and the content of the lexicon. Around 90% of the input words are analyzed and tagged correctly and remaining 10% words are not properly tagged due to spelling variations, compound words and unavailability of the lexical details of the input words in the lexicon. Performance of the proposed system can be improved by updating the details of untagged words into the lexicon.

CONCLUSION AND FUTURE WORK

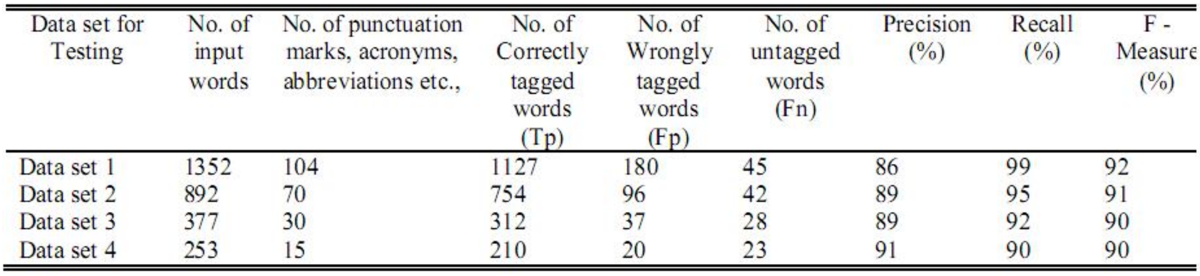
In this proposed work, morphological features and grammatical information of the input words are extracted to determine the parts of speech tags. The Board of Indian standards, Dravidian and hierarchical tag set is used to assign parts of speech tags. It is shown that the performance of morpheme based PoS tagger is better even without using manually pre-tagged training data set and statistical or machine learning algorithms. Since the system is fully linguistic rule governed, the result can be guaranteed to be correct. The overall performance of the proposed system on EMILEE data set is above 90%. The performance is directly proportional to the size of lexicon. Hence, in order to improve the performance, the size of the lexicon can be increased by storing the lexical details with more words into the lexicon. This method can be suitable for other morphologically rich natural languages. The same approach can be extended for chunking, shallow parsing and named entity recognition.

TABLE 4.
HIERARCHICAL TAG SET WITH EXAMPLE



Sl. No.	Top Level	Subtype (level 1)	Subtype (level 2)	Tag Label	Example
1	Noun	Common		NN	Mara
		Proper		NNP	Mysuru
2	Pronoun	Personal		PRP	Naanu
		Reflective		PRF	Thaanu
		Reciprocal		PRC	Paraspara
		Wh-word		PRQ	Yaaru
3	Demonstrative	Diectic		DMD	AA
		Wh-word		DMQ	Yaava
4	Verb	Main	Finite	VF	Hoodanu
			Non-finite	VNF	Hoogi
			Infinite	VINF	Hoogalu
		Auxiliary		VAUX	Hooga beeku
5	Adjective			JJ	suNdaravaada
8	Conjunction			CC	Maththu
9	Particles	Interjection		INJ	Ayyo
		Intensifier		INTF	Thumbaa
10	Quantifiers	Cardinals		QTC	Ondu
		Ordinals		QTO	Ondaneya
11	Residuals	Foreign Word		RDF	Bukku
		Symbol		SYM	@,#,$
		Punctuation		PUNCT	?,.
		Symbol		SYM	@,#,$
		Punctuation		PUNCT	?,.

TABLE 5.
CONFUSION MATRIX – RESULT ANALYSIS OF PROPOSED SYSTEM



Data set for Testing	No. of input words	No. of punctuation marks, acronyms, abbreviations etc.,	No. of Correctly tagged words (Tp)	No. of Wrongly tagged words (Fp)	No. of untagged words (Fn)	Precision (%)	Recall (%)	F - Measure (%)
Data set 1	1352	104	1127	180	45	86	99	92
Data set 2	892	70	754	96	42	89	95	91
Data set 3	377	30	312	37	28	89	92	90
Data set 4	253	15	210	20	23	91	90	90

REFERENCES

[1] E. Brill, "A Simple rule-based part of speech tagger". In Proceedings of the DARPA Speech and Natural Language Workshop. Morgan Kauffman. San Mateo, California, pp. 112-116, 1992.

[2] Ankur Verma and Nitin Hambir,. "Hindi tagger based on transformation rule". International Journal of Computational Linguistics and Natural Language Processing, Vol 2, Issue 3, pp. 293-296, 2013.

[3] Lakshmana Pandian and T. V. Geetha, "Morpheme based language model for Tamil parts of speech tagging", Research journal on computer science and computer engineering with applications, Issue 38, pp. 19-25, 2008.

[4] Merin Francis and Ramachandran Nair, "Hybrid parts of speech tagger for Malayalam". International conference on advances in Computing, communication and informatics, pp. 1744-1750, 2014.

[5] Srinivasu Badugu, "Morphology Based POS Tagging on Telugu", International Journal of Computer Science Issues, Vol. 11, Issue 1, No 1, pp. 181-187, 2014.

[6] B. R. Shambhavi, P. Ramakanth Kumar and G. Revanth, "A maximum entropy approach to Kannada part of speech tagging", International Journal of Computer Applications, Volume 41, No.13, pp. 9-12, 2012.

[7] B. R. Shambhavi and P. Ramakanth Kumar, "Kannada part-of-speech tagging with probabilistic classifiers", International Journal of Computer Applications, Volume 48– No.17, pp. 26-30, 2012.

[8] Pallavi and Anitha S Pillai, "Parts of speech tagger for Kannada using conditional random fields". National Conference on Indian Language Computing, 2014.

[9] Bhuvaneshwari C. Melinamath, "Hierarchical annotator system for Kannada language", Impact: International Journal of Research in Engineering and Technology, pp. 97-110, 2014.

[10] M. C. Padma and R. J. Prathibha, "Development of Morphological Stemmer, Analyzer and Generator for Kannada nouns", Emerging Research in Electronics, Computer Science and Technology, Springer, Vol. 248, pp. 713–723, 2014.

[11] R. J. Prathibha and M. C. Padma, "Development of Morphological Analyzer for Kannada Verbs", Fifth International Conference on Advances in Recent Technologies in Communication and Computing, pp. 22–27, 2013.

[12] H. M. Nayak, "Kannada Rathna Kosha", Kannada Sahithya Parishath, Kannada Abhivruddhi Pradhikara, Bangalore,1994.

[13] Indic NLP library, http://anoopkunchukuttan.github.io/indic_nlp_library/

★★★