Text pre-processing and parts of speech tagging for Kannada language

Saritha Shetty
Department of Master of Computer Applications
NMAM Institute of Technology, Nitte, Karnataka, India
Email- shettysaritha1@nitte.edu.in

Savitha Shetty
Department of Computer Science and Engineering,
NMAM Institute of Technology, Nitte, Karnataka, India
Email- shettysavi1@nitte.edu.in

Abstract - ***The technique of assigning various parts of speech for each word in a text file is referred as Parts of Speech Tagging. This paper propounds assigning of POS for Kannada Language words by Hidden Markov Model. This paper also focuses on Kannada Language detection and Text pre-processing. POS tagger has been developed using Python programming panguage. Tkinter is used as an interface. Data accumulation for training and testing of the system is done from wikipedia, Kannada e-papers. 18000 words are trained and they are tested with 1000 words. The contrast amidst project generated output and physically tagged data results in correctness of accuracy for POS tagging. The correctness of 95% is achieved from the experimental outcome of the proposed System.***

Keywords – ***Hidden Markov Model, Text pre-processing, Lemmatization, Stemming, POS tagging***

I. INTRODUCTION

Language is a medium through which human beings communicate with each other. There are around 7000 languages. Kannada is a language that is preponderantly used by people of Karnataka for communication. It is one among the 1750 languages spoken in India. Over 43.7 million people are the native speakers of this language. There is a special place for Kannada literature in Indian literature. It has a grand history of roughly about two thousand years. The modern alphabets have 49 characters which comprise of 13 vowels, 34 consonants and 2 other speech sounds. The scholarly work related to few of the languages is not present in public realm. Kannada is one of such languages. POS Tagger simply tags the various parts of speech or appropriate tags to the word tokens in a specific sentence or paragraph.

Parts of Speech tagger is used to label tags with the corresponding word tokens. The parts of speech comprise of Preposition, Verb, Interjection, Pronoun, Conjunction, Adjective and Adverb. Natural Language Processing enables the Computers to apprehend, depict and manipulates spoken language. One of the most important applications is POS tagging in Natural Language Processing and it is used to perform other tasks. One word token may have more than one tag. Hybrid approach is used as an approach for tagging the words. This paper presents POS tagging by using the HMM model. Table1 shows the list of tags used in this paper.

TABLE 1 : LIST OF POS TAGS

NO	TAGS	PARTS OF SPEECH	EXAMPLE
1	NN	Noun	ಪರ್ವ
2	NST	Nloc	ಮೇಲೆ
3	NNP	Proper Noun	ಸೀತ
4	PRP	Pronoun	ನನ್ನ
5	DEM	Demonstrative	ಆ
6	VM	Verb Finite	ಬರೆದನು
7	VAUX	Auxiliary Verb	ಬರೆಯುತ್ತಾ
8	JJ	Adjective	ಸುಂದರವಾದ
9	RB	Adverb	ವೇಗವಾಗಿ
10	PSP	Postposition	ಜೊತೆ
11	RP	particles	ಕೂಡ
12	CC	Conjunction	ಮತ್ತು
13	WQ	Question words	ಯಾರು
14	QF	Quantifiers	ಬಹಳ
15	QC	Cardinals	ಒಂದು
16	QO	Ordinal	ಒಂದನೆ
17	CL	Group	ಮಂದಿ
18	COMP	Complimentizer	ಮತ್ತೆ
19	INTF	Intensifier	ತುಂಬಾ
20	INJ	Interjection	ಅಯ್ಯೊ
21	NEG	Negative verbs	ಬಂದಿಲ್ಲ
22	SYM	Symbol	~
23	RDP	Reduplication	ಬೇಗ ಬೇಗ
24	UT	Quotative	ಎಂದು
25	NUM	Numbers	೪೫
26	ECH	Echo words	ಅಕ್ಕಪಕ್ಕ
27	UNK	Unknown	Hello
28	XC	Compounds	ಅಗಿಷ್ಠವರ್ತ
29	CM	Comma	,
30	FS	Full stop	.
31	QM	Question mark	?
32	EXM	Exclamatory mark	!
33	CN	Colon	:
34	SC	Semicolon	;
35	HY	Hyphen	-
36	OB	Opening Bracket	[

37	CB	Closing Bracket	]
38	CF	Closing flower brackets	}
39	OP	Opening Parentheses	(
40	CP	Closing Parentheses	)
41	AP	Apostrophe	'
42	DQ	Inverted Commas	"
43	EL	Ellipse	...
44	AT	At Symbol	@
45	DL	Dollar	$
46	HS	Hash	#
47	VR	Viram	॥

The rest of the paper is organized as follows. Proposed embedding and extraction algorithms are explained in section II. Experimental results are presented in section III. Concluding remarks are given in section IV.

II. LITERATURE SURVEY

POS taggers in different languages have been constructed. Analysis has been carried out for few of the linguistic languages such as Kannada, Telugu, Malayalam, Hindi, Tamil, Bengali, Marathi, Gujarati, Chhattisgarhi, Punjabi, Urdu and Odia.

POS Tagger for Bengali language was proffered using CRF. This tagger used a corpus of over 72,341 words and 26 tags for assessment and efficiency of 90.3% is achieved [1]. POS tagger was proffered for Hindi language using Hidden Markov Model. Pre-processor used is naive stemmer and longest suffix matching algorithm is used. Efficiency achieved is 93.12% [2].

POS Tagger was prefered for Assamese language using Hidden Markov Model. Simple morphological analysis was used to tag unknown words. This tagger used a corpus of over 10,000 words and 172 tags for assessment and correctness of 87% is achieved [3]. POS Tagger was proposed for Hindi Language using Hidden Markov Model. Accuracy of 92% is achieved by using Indian Language POS tag set [4]. POS tagger Malayalam is developed using Conditional Random Field for Malayalam Language, rule based approaches and Support Vector Machine algorithm is used [5]. POS tagger for Kannada Language is proposed using CRF. Data is accumulated from online Kannada newspapers. Corpus of over 1000 words is used for training the system and 45 tags are used for tagging. 99.49% is the accuracy that is achieved by the system [6].

POS tagger for Kannada Language is developed by extracting grammatical information and morphological features of the data. EMILLE corpus is used and the correctness achieved is over 90%[7]. POS tagger for Chhattisgarhi language is developed by using rule based parts of speech tagger. Corpus of over 40,000 is used for training the data and 30 tags are used for tagging the words. The correctness of the system achieved is 78%[8].

II. METHODOLOGY

The proposed System is constructed using Hybrid approach. Hidden Markov Model is used for tagging the word with appropriate tags. The input given is Kannada Text File. At first, the Text file is split into sentences and tokens with the help of tokenizer. The language detection is done to make sure that only Kannada words are taken into consideration. The language detection is done by checking if the data is within the Unicode range. If the word is not in Kannada then it will assign UNK tag to the tagged data and then display the tagged data. In the proposed system, punctuations are also considered as separate tags.

2.1 Language detection –

Kannada Language is detected by using Unicode range of Kannada characters. If the Unicode of the data is within the range of U+0C80...U+0CFF then it is considered as a Kannada word.

2.2 Text pre-processing –

Text pre-processing benefits us in cleaning the text before further processing of the data. The assorted pre-processing techniques used are:

1) Removal of Stop words or the words those are not so important in the text and filtering the data
2) Removal of punctuation marks which are not so useful
3) Tokenization is performed for sentences and tokens to simplify the further processing of data.
4) Stemming and lemmatization

Lemmatization is a method of transforming the words in a sentence to the dictionary form of that word. Stemming is a method of lowering words to their respective word stem. The word stems are not required to be of same root word as their morphological root in the dictionary. The word can be equal or can be it's smaller form. In Stemming few predefined prefixes and suffixes are checked in the words and then it is curtailed to its normal form. In Lemmatization, Tokens are curtailed to its essence. A dictionary is maintained to change the altered form into its base form.

2.2 Hidden Marchov Model –

The algorithm applied is HMM model for projected parts of speech tagger. It is a statistical model to be used to interpret the observable events that are depended on the internal factors, which are hidden and they are not observable. The hidden states in our proposed system are tags and observable states are words.

1. Emission matrix is of size : [tags]*[words]
   Emission matrix depicts the probabilities of forming valid observations when a specific state is given.

2. Transition matrix is of size [tags]*[tags].
   Transition matrix depicts the probabilities of changing from a specific state to some other state. During each transmission least used trellis path is eliminated.

Dataset preparation : The tagging of words are done by tagging tags from user-defined tag set. The proposed system contains 48 tags. The data is accumulated from Wikipedia and e-papers. Few moral stories from Kannada Language are also taken into consideration.

2.2.1 Algorithm for implementation of HMM

Input : Untagged data as input
Output : Tagged data

Step 1 : Kannada sentence and words are split through line splitter code.
Step 2 : Duplicate words are discarded and count of the tag occurrence is calculated.
Step 3 : Emission and Transition matrix is computed.
Step 4 : Open Kannada test file
Step 5 : For range in length(tags)
Step 6 : Find the maximum probability by using Viterbi Algorithm. Go back to step5
Step 7 : The tagged data corresponding to words are displayed.

2.2.2 Viterbi algorithm

Step 1: Initialization - Initialize the viterbi matrix of the size |no_of_tags|*|no_of_words_in_test_sentence|
Step 2 : Updation of the viterbi matrix - viterbi decoder is used in order to update the matrix. All possibilities of hidden states are calculated.
Step 3: Select the most suitable tag for the last word of the test sentences by identifying the maximum probability for the last column.
Step4 : Final step - Identify the most suitable tag for every word by backtracking.

IV. RESULTS AND DISCUSSION

Correctness of the proposed system is checked by comparing human annotated tags with the system generated output. We compare correctly tagged words with the total number of tags that are present to find the accuracy. The proposed system attains correctness of 95%.

$$Accuracy of the system = \frac{total number of tokens correctly tagged}{total number of words}$$

The text pre-processing is carried out to clean the text so that it is ready to be processed by the machine.

Lemmatization and Stemming :

TABLE 2: EXAMPLES FOR LEMMATIZATION AND STEMMING

Word	Stem	Lemma
ಹೂವಿನಿಂದ	ಹೂವಿ	ಹೂವು
ಮಾತನ್ನು	ಮಾತ	ಮಾತು
ಪ್ರಯತ್ನಿಸು	ಪ್ರಯತ್ನಿ	ಪ್ರಯತ್ನ

Table 2 shows that lemmatization is analogously superior to stemming. The altered word is curtailed to its essence making it simple to be handled. In the projected system, the HMM model is used to tag the POS tags with the Kannada text. Thousand distinct words are used for measuring the system.

ಭಾರತದ ಕರಾವಳಿಯು ೭೫೧೭ ಕಿಲೋಮೀಟರ್ (4700 ಮೈಲಿ) ಉದ್ದವನ್ನು ಹೊಂದಿದೆ;

ಭಾರತದ_NNP ಕರಾವಳಿಯು_NN ೭೫೧೭_NUM ಕಿಲೋಮೀಟರ್_VM (_OP 4700_UNK ಮೈಲಿ_NN)_CP ಉದ್ದವನ್ನು_NN ಹೊಂದಿದೆ_VM ;_SM ._.

Figure 1: Output of POS Tagging

Text pre-processing is done independently in the proposed system because even punctuations are considered as separate tags. Output of POS tagging is shown in Figure 1.

IV. CONCLUSION

In natural language processing POS tagging is very much essential. Parts of Speech tagger in Kannada language by making use of HMM model has been conferred. Tokenization of sentences is done by line splitter program. These words are then checked by the probability of the emission and transition matrix and the appropriate tags are assigned. The proposed system is developed with corpus size of 18000 words with tag set of 48 parts of speech tags. Around thousand words are used as test data.

REFERENCES

[1] Ekbal, A., Haque, R., & Bandyopadhyay, S. (2007, December). Bengali part of speech tagging using conditional random field. In *Proceedings of Seventh International Symposium on Natural Language Processing (SNLP2007)* (pp. 131-136).

[2] Shrivastava, M., & Bhattacharyya, P. (2008, December). Hindi POS tagger using naive stemming: harnessing morphological information without extensive linguistic knowledge. In *International Conference on NLP (ICON08), Pune, India*.

[3] Saharia, N., Das, D., Sharma, U., & Kalita, J. (2009, August). Part of speech tagger for Assamese text. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers* (pp. 33-36). Association for Computational Linguistics.

[4] Joshi, N., Darbari, H., & Mathur, I. (2013). HMM based POS tagger for Hindi. In Proceeding of 2013 International Conference on Artificial Intelligence, Soft Computing (AISC-2013).

[5] Francis, M., & Nair, K. R. (2014, September). Hybrid part of speech tagger for Malayalam. In *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (pp. 1744-1750). IEEE.

[6] Pallavi, A. S. P., & Pillai, A. S. (2014). Parts Of Speech (POS) Tagger for Kannada Using Conditional Random Fields (CRFs). In *Proceedings of the National Conference on Indian Language Computing, NCILC*.

[7] Padma, M. C., & Prathibha, R. J. (2016). Morpheme based parts of speech tagger for Kannada language. The IRES International Conference, Los Angeles.

[8] Pandey, V., Padmavati, M. V., & Kumar, R. Rule Based Parts of Speech Tagger for Chhattisgarhi Language.

[9] Reddy, M. V., & Hanumanthappa, M. (2012). POS Tagger for Kannada Sentence Translation. *Int. J. Emerg. Trends Technol. Comput. Sci. (IJETTCS)*, *1*, 490.

[10] Mehta, P., & Majumder, P. (2016). Large scale quantitative analysis of three Indo-Aryan languages. *Journal of Quantitative Linguistics*, *23*(1), 109-132.

[11] Choudhry, A. (1995). Models of Bilingual measurement and their adaptability in the Indian context. *Journal of Quantitative Linguistics*, *2*(3), 258-266.