International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 9 Issue 7 ǁ 2021 ǁ PP. 13-18

Development of Stochastic Part Of Speech Tagger for Morphologically Rich Languages

Gurmit Kaur
Research Scholar
DeshBhagat Foundation Group of colleges Dagru (Moga)

Deepak Sharma
Assistant Professor
DeshBhagat Foundation Group of colleges Dagru (Moga)

***Abstract**: Natural language processing is one of the most emerging field in computer science research. In this research various applications on language processing are developed. Part of a speech tagging also called POS tagging is one of the most important component in almost all the Natural language Processing applications. A lot of efforts are being done by various researchers to improve the efficiency of the part of speech tagger. Further, development of POS tagger for morphologically rich languages is again a challenging task. In this research paper we have developed a part of speech tagger for one of the morphologically rich language i.e. Punjabi language. We have used n-gram stochastic method for its development of this tagger. On testing this system and comparing the results it is observed that this method gives a better result as compare to rule based method. Also this POS tagger does not require any linguist knowledge.On testing the developed POS tagger author claimed precision as 93.86, recall as 94.92 and f-measure as 94.3.*
***Keywords**: POS tagger, Punjabi, morphological rich language.*

Date of Submission: 10-07-2021 Date of acceptance: 26-07-2021

I. Introduction

Various efforts are being done by various researchers for technical development of Punjabi language [32-35]. The problem of tagging in natural language processing is to find a way to tag every word in a text as a particular part of speech, e.g., proper pronoun. POS tagging is a very important preprocessing task for language processing activities.Part of Speech tagging is a process of marking the words in a text as corresponding to a particular part of speech, based on its definition, as well as its context. POS tagging is a very important preprocessing task for language processing activities. This helps in doing deep parsing of text and in developing Information extraction systems, semantic processing etc. POS tagging for natural language texts have been developed using linguistic rule, stochastic models and a combination of both.Part of Speech (POS) taggers have been developed for various Indian Languages like Hindi, Punjabi, Malayalam, Bengali and Telugu. Various part of speech tagging approaches like N-gram, Hidden Markov Model (HMM), Support Vector Model (SVM), Rule based approaches, Maximum Entropy (ME) and Conditional Random Field (CRF) have been used for POS tagging. Accuracy is the prime factor in evaluating any POS tagger.

Different POS tagging techniques
There are many different techniques used for development of part of speech taggers. These techniques can be classified into two categories i.e. supervised POS tagging and Unsupervised POS tagging.
Supervised tagging: **-**This method is based on pre-tagged corpora. It is amethod of facilitating in the system of disambiguation or to learn the rules for tagging.
Unsupervised tagging: - This method on the other hand do not require pre-tagged corpus. The unsupervised POS Tagging models do not require a pre-tagged corpus. Instead, they use advanced computational techniques like the Baum-Welch algorithm to automatically induce tagsets, transformation rules, etc. Based on this information, they either calculate the probabilistic information needed by the stochastic taggers or induce the contextual rules needed by rule based systems or transformation based systems. They are further two divided into two distinct approaches for POS Tagging-Rule based and Stochastic approaches. Rule based approach uses a large database of hand-written disambiguation rules considering the morpheme ordering and contextual information. The Stochastic approach uses an unambiguously tagged text to estimate the probabilities to select the most likely sequence. For selecting the maximum likelihood probability the lexical generation probability and the n-gram probability are considered. The most common algorithm for implementing an n-gram approach

is the Viterbi Algorithm which follows a Hidden Markov Model.

II. Literature review:

As discussed in above section, various POS tagging techniques are used by different researchers to develop part of speech tagger. In case of Punjabi language rule based approach is used by Singh M. et.al. (2008). If we talk about Hindi language then early work started with development of the partial POS tagger by Ray et.al [2]. Further Shrivastava et al. proposed harnessing morphological characteristics of Hindi for POS tagging [3]. This was further enhanced in [4], which suggests a methodology that makes use of detailed morphological analysis and lexicon lookup for tagging. The accuracy was 93.45% with a tagset of 23 POS tags. Further International Institute of Information Technology (IIIT), Hyderabad, initiated a POS tagging and chunking contest, NLPAI ML for the Indian languages in 2006. Several teams came up with various approaches for tagging in three Indian languages namely, Hindi Bengali and Telugu. In this contest, CRFs were first applied to Hindi by Ravindranet. Al. [5] and Himanshuet. al.[6] for POS tagging and chunking, where they reported a performance of 89.69% and 90.89% respectively. In the work of SankaranBhaskaran [7], HMM based statistical technique was attempted. Here probability models of certain contextual features were also used. POS tagging of Hindi language based on Maximum Entropy Markov Model was developed by AniketDalal et al [8]. In this system, the main POS tagging features used were context based features, dictionary features, word features, and corpus-based features.In 2007, as part of the SPSAL workshop in IJCAI-07, IIIT, Hyderabad conducted a competition on POS tagging and chunking for south Asian languages of Hindi, Bengali and Telugu. The average POS tagging accuracy of all the developed systems for Hindi, Bengali and Telugu are 73.93 %, 72.35 % and 71.83 % respectively.ManishShrivastava&Pushpak Bhattacharyya [9] designed a simple POS tagger for Hindi based on HMM. It utilized the morphological richness of the language without restoring to complex and expensive analysis. It achieved a good accuracy of 93.12%. Recent work in this area has been one by Ankur Parikh [10] where Neural Networks are tried for tagging. In case of Bengali language participants at NLPAI Contest 2006 and SPSAL 2007 tried tagging for Bengali along with Hindi and Telugu. The highest accuracies obtained were 84.34% and 77.61% for Bengali in the contests respectively. HMM based tagger is reported in [11]. Maximum Entropy based tagger was built in [12]. This tagger demonstrated an accuracy of 88.2% for a test set of 20,000 word forms. CRF and SVM based taggers are reported in [13] and [14] respectively. SVM tagger used 26 tags and had a performance of 86.84%.Recently Ekbalet. al applied voted approach [15] in order obtain best results in Bengali tagging. Further in case of Tamil a work by VasuRanganathan named tag tamil is based on Lexical phonological approach. The tagger does morph tactics of morphological processing of verbs by using index method. Ganeshan's POS Tagger [16] works on CIIL corpus. The tagset includes 82 tags at morph level and 22 at word level. Kathambam is a heuristic rule based tagger designed at RCILTS-Tamil. The performance of the tagger is around 80%. It is based on the bigram model. In [17] a hybrid tagger using rule based and HMM technique is developed. SVMTool was used to tag the corpus in [18] and an accuracy of 94.12% was obtained. LakshmanaPandian and Geetha [19] experimented with a morpheme based tagger. A naive Bayes probabilistic model using morphemes is the first stage for preliminary POS tagging and a CRF model is the next stage to disambiguate the conflicts that arise in the first stage. The overall accuracy of the tagger was 95.92%. Dhanalakshmi et.al [20] used SVM methodology based on Linear programming. This gave the accuracy of 95.63% on the test data. POS tagger for Telugu was developed by Sreeganesh [21] using a rule based approach. In the initial stage, a Telugu Morphological Analyzer analyses the input text. During NLPAI Contest 2006, a POS tagger of accuracy 81.59%was built. In SPSAL 2007 workshop of IJCAI-07, the best Telugu tagger was proposed by Avineshet. al [22] with a performance of 77.37%. In [23], three Telugu taggers namely (i) Rule-based tagger, (ii) Brill Tagger and (iii) Maximum Entropy tagger were developed with accuracies of 98.016%, 92.146%, and 87.81% respectively. Recent work has been by SindhiyaBinulalet. al [24] who applied SVMTool to tagging. First POS tagger for Gujrati was developed byChirag Patel andKarthikGali [25] using a hybrid model. An accuracy of 92% has been achieved by this approach. For Malyalam, Manju K et. al [26] experimented with the stochastic approach for tagging of Malayalam words. The results obtained were promising. Later work was by Antony P.J et. al [27] who applied SVM approach to tag words. With the increase in the number of words in the training set, the performance increased to around 94%. In case of Manipuri language ThoudamDoren Singh and SivajiBandyopadhyay initially tried to build a morphology driven tagger [28]. This showed an accuracy of only 69%. Later they built a tagger [29] using Conditional Random Field (CRF) and Support Vector Machine (SVM). The tagset consisted of 26 tags. Evaluation results demonstrated improvement in the accuracies. They obtained 72.04%, and 74.38% accuracies in the CRF, and SVM, respectively. In case of NavanathSaharia et.al [30] built first Assamese tagger using the HMM model with Viterbi algorithm. An accuracy of 87% was achieved by the tagger for the test inputs. Pallav Kumar Dutta has attempted to develop an online semi-automated tagger. This was designed to deal with sparse data problem of the language. NLTK is used to tag the test data and for the ambiguous tags an online tagger would help the user to change the tags.

III. Methodology Used:

In this research we used n gram based probability for developing the part of speech tagger. In n-gram we used bi-gram. Further to generate bi-grams, annotated corpus of Punjabi language is used. Now since no standard annotated corpus is available for Punjabi language, therefore we created our own annotated corpus. The steps followed for creation of annotated corpus are displayed in figure 1.

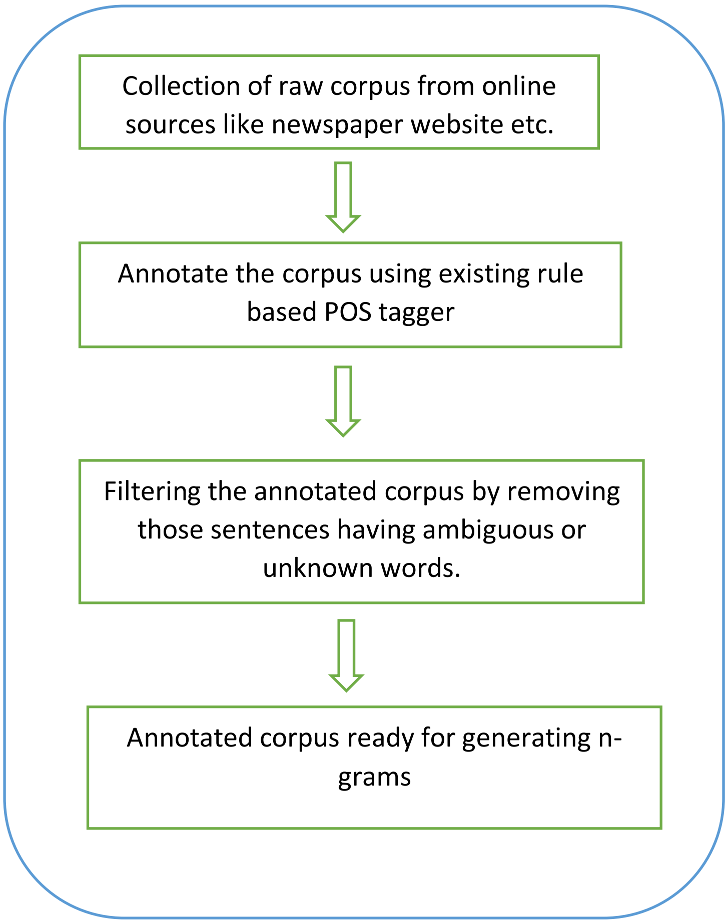

Figure 1: creation of annotated corpus

The details of the annotated data generated is tabulated in table 1.

Table 1: details of the annotated data generated

Sr.no.	Source of corpus collection	Number of sentences collected	Number of sentences having unknown words	Number of sentences having ambiguous words	Number of annotated sentences available to generate bi-grams
1		4500	897	165	3438
2		5000	1323	87	3590
3		5000	1432	92	3476
4		7000	1276	142	5582
5		4500	1076	89	3335
Total		26000	6004	575	19421

After generating annotated corpus, bi-gram probabilities of part of speech tags are generated. To generate these bi-gram probabilities, first the word and tags were separated i.e. only tag pattern is generated. This tag pattern is generated by removing the word from the corpus and joining the tags sentence-wise. Some sample tag entries are shown in table 2.

Table 2: some sample tag pattern entries

Sr. No	Tag patterns
1	AJIFPO_ NNFPO_ PPIBSD_ VBMAMPXXPINIA_ NNMSO_ PPIMPD_VBMAFPXXXINDA_ VBAXBPT1_ Sentence
2	NNMSD_PTUE_NNMPD_AJIMSO_NNMSO_PPIMPD_Comma_AJIMPD_CJC_NNFPD_VBMAMPXXXINDA_ VBAXBPT1_ Sentence
3	PPIFSO_ NNFSO_ PPIMSO_ NNMPO_ PTUE_ NNMPD_ AJIMPD_ CJC_ AJU_ VBP_ VBMAMPXXXTNDA_ VBAXBPT1_ Sentence
4	CJU_ Unknown_ PPIFSD_ NNBSD_ AJU_ NNMSD_ VBMAMPXXPINIA_ PTUE_ PPIFPD_ NNFPD_ AVU_ VBMAFPXXXINNA_ VBP_ VBMAXSS3XINO_ VBMAFPXXXINDA_ VBAXBPT1_ Sentence
5	AJU_ PTUE_ AJIMPD_ NNMPD_ VBMAXSS3XTNO_ VBMAMPXXXINDA_ VBAXBPT1_ Sentence
6	NNMSO_ PPIMPD_ CDPA_ Hyphen_ CDPA_ NNMPO_ PPIMSO_ NNMPO_ PPU_ NNFPO_ PPIBSD_ Sentence
7	NNMSD_ CDPA_ Hyphen_ CDPA_ NNMPO_ AVU_ CJC_ AJIMPD_ AJIFSO_ NNFSO_ PPIMSO_ NNMPO_ PPU_ CDPA_ Hyphen_ CDPA_ NNMPO_ AVU_ VBMAXPSXXTNE_ Sentence
8	AJIFSD_ CJC_ AJU_ NNFSD_ VBMAFSXXXTNIDA_ VBAXBST1_ Sentence
9	CJU_ NNFSD_ PTUE_ CJC_ AJIFSD_ PTUE_ NNMSD_ Hyphen_ NNMSO_ PPIBSD_ NNMPO_ PPIFSO_ NNFSO_ PPU_ NNMSD_ VBMAXPSXXTNE_Sentence
10	NNMSO_ PPIBSD_ CJU_ NNFSD_ PTUE_ CDPA_ Hyphen_ CDPA_ VBMAFPXXXTNNA_ VBMAFPXXXTNIDA_ VBAXBPT1_ Sentence

After generation of tag pattern, probability of bi-gram is calculated from following formula:

$$P_{(bi\text{-}gram)} = \frac{Number\ of\ times\ a\ bigram\ appears\ in\ the\ POS\ pattern}{Total\ number\ of\ bi-grams}$$

From above formula it is clear that probability of a bi-gram is calculated as number of times that bi-gram appears in the tag pattern corpus divided by total number of unique bi-grams generated from tag patterns. Some sample entries of bi-gram probability is tabulated in table 3.

Table 3: some sample entries of bi-gram probabilities

Sr. No.	Bi-gram	Probability
1	NNFSD_VBP	0.0457059206245934
2	VBP_VBMAXSS3XBNO	0.00201072386058981
3	VBMAXSS3XBNO_PTUKE	0.0106382978723404
4	PTUKE_PNPMPGDF 0.0024	0.0024
5	PNPMPGDF_NNMSO	0.00289181220231345
6	NNMSO_PPIBSD	0.117528483786152
7	PPIBSD_NNMSD	0.00675626412618174
8	NNMSD_VBMAMSXXPINIA	0.0313037865748709
9	VBMAMSXXPINIA_CJC	0.00504964053406368
10	CJC_AVU	0.0113620569840167

Algorithm Used:
Step1: Input Punjabi sentence in Unicode format.
Step2: Apply Morphological Analyzer to make it annotated sentence.
Step3: From the annotated sentence created in step2, create consecutive tag pairs.
Step4: from the tag pairs created in step 3, identify the tag pairs having ambiguous tags.
Step5: from the tag pairs having ambiguous tags, generate all possible combinations of tag pairs.
Step6: assign the pre-calculated bigram probabilities to each pair generated in step5.
Step7: pair having maximum probability will be selected and all other combinations will be discarded.
Step 8: Assign the selected pair in step7 to the corresponding words.

IV. Results And Discussion:

Author tested this system on test data taken from the annotated corpus developed. Actually 70% of the annotated corpus generated was used to generate bi-grams and remaining 30% of the annotated corpus was used to test the system. The results obtained after applying these bi-gram probabilities on test corpus is shown in table 4. Further, when same test data is applied on existing POS tagger, the results obtained are tabulated in table 5.

Table 4: Results obtained on testing the developed system

Sr.No.	Number of sentences in test data	Number of ambiguities in sentences	Number of ambiguities correctly handled by developed system	Number of ambiguities in-correctly handled by developed system	Number of ambiguities not handled by the developed system	Precision	Recall	F-measure
1	165	192	180	10	2	93.7	94.7	94.2
2	87	110	102	8	0	92.7	92.7	92.7
3	92	125	117	6	2	93.6	95.1	94.3
4	142	162	152	9	1	93.8	94.4	94.1
5	89	113	108	3	2	95.5	97.2	96.3
Average						93.86	94.82	94.3

Table 5: Results obtained on testing the existing system

Sr. No.	Number of sentences in test data	Number of ambiguities in sentences	Number of ambiguities correctly handled by existing system	Number of ambiguities in-correctly handled by existing system	Number of ambiguities not handled by the existing system	Precision	Recall	F-measure
1	165	192	178	12	2	92.7	93.7	93.2
2	87	110	98	10	2	89.1	90.7	89.9
3	92	125	111	10	4	88.8	91.7	90.2
4	142	162	145	12	5	89.5	92.4	90.9
5	89	113	107	3	3	94.7	97.3	96.0
Average						93.2	91.0	92.0

Conclusion and future scope: In this research work, author attempted to develop a part of speech tagger for morphologically rich Punjabi language. Author used n-gram based stochastic probability based approach. Further to create bigram probabilities, corpus from online resources is collected. From the results shown in table 4 and table 5, it can be concluded that the developed part of speech tagger performs better as compare to existing rule based POS tagger.On testing the developed POS tagger author claimed precision as 93.86, recall as 94.92 and f-measure as 94.3. The results obtained are better as compare to rule based systems which on testing on the same data shows precision as 93.2, recall as 91.0 and f-measure as 92.0. Now since the developed POS tagger is independent of language and therefore, in future this system can be extended to be used for other morphologically rich languages by just changing the corpus.

REFERENCES

[1] Ahmed (2002), "Application of multilayer perceptron network for tagging parts-of-speech", Proceedings of the Language Engineering Conference, IEEE.

[2] A. Basu P. R. Ray, V. Harish and S. Sarkar(2003), "Part of speech tagging and local word grouping techniques for natural language parsing in Hindi", Proceedings of the International Conference on Natural Language Processing (ICON 2003).

[3] S. Singh M. Shrivastava, N. Agrawal and P. Bhattacharya (2005), "Harnessing morphological analysis in pos tagging task", Proceedings of the International Conference on Natural Language Processing (ICON 2005).

[4] Smriti Singh, Kuhoo Gupta, Manish Shrivastava, and Pushpak Bhattacharyya (2006),"Morphological richness offsets resource demand – experiences in constructing a pos tagger for Hindi", Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, Sydney, Australia, pp. 779–786.

[5] PranjalAwasthi, Delip Rao, BalaramanRavindran (2006), "Part Of Speech Tagging and Chunking with HMM and CRF", Proceedings of the NLPAI MLcontest workshop, National Workshop on Artificial Intelligence.

[6] Himanshu Agrawal, Anirudh Mani (2006), "Part Of Speech Tagging and Chunking Using Conditional Random Fields" Proceedings of the NLPAI MLcontest workshop, National Workshop on Artificial Intelligence.

[7] SankaranBaskaran (2006), "Hindi POS tagging and Chunking", Proceedings of the NLPAI MLcontest workshop, National Workshop on Artificial Intelligence.

[8] AniketDalal, Kumar Nagaraj, Uma Sawant, Sandeep Shelke (2006), "Hindi Part-of-Speech Tagging and Chunking: A Maximum Entropy Approach" Proceedings of the NLPAI MLcontest workshop, National Workshop on Artificial Intelligence.

[9] Manish Shrivastava, Pushpak Bhattacharyya (2008), “Hindi POS Tagger Using Naive Stemming: Harnessing Morphological Information Without Extensive Linguistic Knowledge”, Proceedings of ICON-2008: 6th International Conference on Natural Language Processing.
[10] Ankur Parikh (2009), “Part-Of-Speech Tagging using Neural network”, Proceedings of ICON-2009: 7th International Conference on Natural Language Processing.
[11] Ekbal, Asif, Mondal, S., and S. Bandyopadhyay (2007) “POS Tagging using HMM and Rule-based Chunking”, In Proceedings of SPSAL-2007, IJCAI-07, pp. 25-28.
[12] A. Ekbal, R. Haque and S. Bandyopadhyay (2008), “Maximum Entropy Based Bengali Part of Speech Tagging”, Advances in Natural Language Processing and Applications, Research in Computing Science (RCS) Journal, Vol. (33), pp. 67-78.
[13] A. Ekbal, R. Haque and S. Bandyopadhyay (2007), “Bengali Part of Speech Tagging using Conditional Random Field”, Proceedings of the 7th International Symposium on Natural Language Processing (SNLP-07), Thailand, pp.131-136.
[14] A. Ekbal and S. Bandyopadhyay (2008), “Part of Speech Tagging in Bengali using Support Vector Machine”, Proceedings of the International Conference on Information Technology (ICIT 2008), pp.106-111, IEEE.
[15] A. Ekbal , M. Hasanuzzaman and S. Bandyopadhyay (2009), “Voted Approach for Part of Speech Tagging in Bengali”, Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation (PACLIC-09), December 3-5, Hong Kong, pp. 120-129.
[16] Ganesan M (2007), “Morph and POS Tagger for Tamil” (Software) Annamalai University, Annamalai Nagar.
[18] Arulmozhi P, Sobha L (2006) “A Hybrid POS Tagger for a Relatively Free Word Order Language”, Proceedings of MSPIL-2006, Indian Institute of Technology, Bombay.Dhanalakshmi V, Anandkumar M, Vijaya M.S, Loganathan R, Soman K.P, Rajendran S (2008), “Tamil Part-of-Speech tagger based on SVMTool”, Proceedings of the COLIPS International Conference on Asian Language Processing 2008 (IALP), Chiang Mai, Thailand.
[19] S. LakshmanaPandian and T. V. Geetha (2008), “Morpheme based Language Model for Tamil Part-of-Speech Tagging”, Research journal on Computer science and computer engineering with applications, July-Dec 2008, pp. 19-25.
[20] Dhanalakshmi V, Anandkumar M, Shivapratap G, Soman, K P, Rajendran S (2009) “Tamil POS Tagging using Linear Programming”, International Journal of Recent Trends in Engineering, 1(2) pp.166-169.
[21] T. Sreeganesh(2006), “Telugu Parts of Speech Tagging in WSD”, Language of India, Vol 6: 8 August 2006.
[22] Avinesh PVS and KarthikGali (2007), “Part-of-speech tagging and chunking using conditional random fields and transformation based learning”, Proceedings of the IJCAI and the Workshop On Shallow Parsing for South Asian Languages (SPSAL), pp. 21–24.
[23] Rama Sree, R.J, KusumaKumari P (2007), “Combining POS Taggers for improved Accuracy to create Telugu annotated texts for Information Retrieval”, Tirupati.
[24] G.SindhiyaBinulal, P. AnandGoud, K.P.Soman(2009), “A SVM based approach to Telugu Parts Of Speech Tagging using SVMTool”, International Journal of Recent Trends in Engineering, Vol. 1, No. 2, May 2009
[25] Chirag Patel and KarthikGali (2008), “Part-Of-Speech Tagging for Gujarati Using Conditional Random Fields”, Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages, Hyderabad, India, pp. 117–122.
[26] Manju K, Soumya S, Sumam Mary Idicula (2009), “Development of A Pos Tagger for Malayalam-An Experience”, Proceedings of 2009 International Conference on Advances in Recent Technologies in Communication and Computing, IEEE
[28] Antony P.J, Santhanu P Mohan, Soman K.P (2010), “SVM Based Part of Speech Tagger for Malayalam”, Proceedings of 2010 International Conference on Recent Trends in Information, Telecommunication and Computing, IEEE.
[29] ThoudamDoren Singh, SivajiBandyopadhyay (2008), “Morphology Driven Manipuri POS Tagger”, Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages, Hyderabad, India, pp. 91–98.
[30] ThoudamDoren Singh, SivajiBandyopadhyay (2008), “Manipuri POS Tagging using CRF and SVM: A Language Independent Approach”, Proceedings of ICON-2008: 6th International Conference on Natural Language Processing.
[31] NavanathSaharia, Dhrubajyoti Das, Utpal Sharma, JugalKalita (2009), “Part of Speech Tagger for Assamese Text”, Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, Suntec, Singapore, pp. 33–36.
[32] Sharma, Sanjeev Kumar, and Gurpreet Singh Lehal. "Using Hidden Markov Model to improve the accuracy of Punjabi POS tagger." 2011 IEEE International Conference on Computer Science and Automation Engineering. Vol. 2. IEEE, 2011.
[33] Gill, Mandeep Singh, Gurpreet Singh Lehal, and Shiv Sharma Joshi. "Part of speech tagging for grammar checking of punjabi." The Linguistic Journal 4.1 (2009): 6-21.
[34] Mittal, Sumeer, Navdeep Singh Sethi, and Sanjeev Kumar Sharma. "Part of speech tagging of Punjabi language using N gram model." International Journal of Computer Applications 100.19 (2014).
[35] Kaur, Manjit, MehakAggerwal, and Sanjeev Kumar Sharma. "Improving Punjabi Part of Speech Tagger by Using Reduced Tag Set." International Journal of Computer Applications & Information Technology 7.2 (2014): 142.