An Approach for Grammatical Constructs of Sanskrit Language using Morpheme and Parts-of-Speech Tagging by Sanskrit Corpus

Namrata Tapaswi  
NIMS University, Jaipur, Raj., India

S.P. Singh  
NIMS University, Jaipur, Raj., India

Suresh Jain  
NIMS University, Jaipur, Raj., India

Abstract-Sanskrit since many thousands of years has been the oriental language of India. It is the base for most of the Indian Languages. Statistical processing of Natural Language is based on corpora (singular corpus). Collection of texts of the written and spoken words is known as Language corpus, which is collected in an organized way, in electronic media for the purpose of linguistic research. It presents as a resource to be systematically ‘consulted’ by language investigators. This paper explains an approach for tagging the corpora automatically at word and morphemic levels for Sanskrit. It also gives different tag sets used at both the levels.

Keywords- Part-Of-Speech, tagging, noun, verb, parsing, lexical analysis.

I. INTRODUCTION

Understanding of actual sense of word is very tricky. Most of the words having more than one meanings like in English language word “book” plays two different role in the sentences i.e., Book that flight and this is a book. two major approaches to deal with the problem of sense disambiguation of words. The knowledge-based approach uses explicit sets of lexicon , while the corpus-based approach uses information obtained from corpus. As we prefer to work with the corpus based approach, we try to extract information from the analysis of corpus. Information which we get will be processed to understand actual contextual sense.

Besides linguists and lexicographers potentiality of large corpora (in English and other European Languages) has been identified by speech and information technologists. Again corpora are analyzed, and the linguistic information is enclosed at various levels (tagged corpora). Automatically retrieve selective information for the convenience of researchers to. The various sectors where corpora are found to be functional are linguistics, lexicography, natural language processing, language teaching and speech processing.

II. LITRETURE REVIEW

‘Development of Corpora of text of Indian Languages’ started in1991 by Department of Electronics (DOE), Govt. of India for the first time the texts of Indian languages are made available in machine readable form through the project. Six various sectors at which, corpora development project for the 15 scheduled languages has been chosen. For formation of corpus, later languages newly added to the 8th schedule have also been added. Elaborately by Annamalai (1994) have been discussed objective, size of the corpora, coordination between centers, etc.. The Central Institute of Indian Languages, Mysore has taken up the corpora development work for Sanskrit, Kannada, Malayalam, Tamil and Telugu Languages. This paper explains an approach for tagging the corpora automatically at word levels and morphemic levels for Sanskrit. It also gives different tag sets used at both the levels.

Various studies have been done for Part-Of-Speech tagging, Dinesh Kumar and Gurpreet Singh Josan suggests prime factor in evaluating any POS tagger [1].Dipanjan Das,Slav Petrov introduced unsupervised part-of-speech taggers for languages that have no labeled training data, but have translated text in a resource-rich language[2].Doug
Cutting, Julian Kupiec, Jan Pedersen, Penelope Sibun introduced part-of-speech tagger is a system that uses context to assign parts of speech to words[3]. B. Megyesi introduced In Hungarian language; it is shown that the present system does not obtain as high accuracy for Hungarian as it does for English [4]. Mitchell P. Marcus, Beatrice Santorini and Mary A. Marcinkiewicz proposed First design POS tagset and presenting the tagset itself, after that two-stage tagging process, in which text is first assigned POS tags automatically and then corrected by human annotators[5]. Cem Bozsahin described lexicon to formulate semantically transparent specifications[6]. Namrata Tapaswi and Dr. Suresh Jain introduced how to morph the Sanskrit sentences[7]. Evangelos Dermatas, George Kokkinakis described stochastic tagger that are able to predict POS of unknown words [8]. Doug Cutting, Julian Kupiec described implementation strategies and optimizations which result in speed high speed operation[9]. Mitchell P. Marcus, Beatrice Santorini and Mary A. Marcinkiewicz described how to constructing one such large annotated corpus—the Penn Treebank [10].

We qualitatively analyze our results by examining the categorization of several high impact papers. With consultation from prominent researchers and textbook writers in the field, we propose simple corpus tagging for Sanskrit language. It uses rule based approach to tag each word of the sentence.

III. CORPUS MANAGEMENT

Corpus contains a number of various information rather then texts which, in turn, makes the information retrieval a relatively trivial task. Mainly this Information’s are categorized in two types:
1) Representative information (actual form of the text) and
2) Interpretative information (adding linguistic information to the text).

IV. CORPUS ANNOTATION

Individual sentences, words, morphemes, etc, Corpus can be made to provide more valuable information about these. It could be gained by including linguistic information (interpretative information) to the text. The electronic representation of language material itself is called corpus annotation including the practice of adding linguistic information to an existing corpus of spoken or written language by some kind of coding attached to, or interspersed with. Annotations can be made at different levels, namely, orthographic, phonetic/phonemic, prosodic, grammatical, syntactic, semantic and pragmatic/discourse level. Basic advantages of annotated corpora are that the structural information at various levels could be retrieved based on linguistic tags, which are the frequent requirements of linguists, lexicographers and NLP researchers.

V. GRAMMATICAL TAGGING

In English and other European languages Grammatical tagging is the popular and common type of annotation successfully implemented in a number of corpora. It is the procedure, to indicate its grammatical category, which adds a tag at the end of a word. It can be achieved in two ways:
1) Manual tagging and
2) Automatic tagging (with manual post-editing).

The former is labor intensive, slow and legally responsible for error and inconsistency. There are various approaches in the latter, but can be broadly categorized in two methods:

1) Rule based tagging –
These taggers are based on a defined set of hand written rules. Most of the existing Rule Based POS taggers are based on two-stage architecture. The first stage assigns a list of probable tags (or the basic tag) for a particular word. The second stage, uses large list of hand written disambiguation rules, to reduce the list (or change a wrong tag) to a single right tag. Here all the rules are pre-defined. They may be language dependent or independent.

2) Statistics based tagging –
Stochastic taggers used hidden Markov Model or HMM tagger. The purpose behind all stochastic taggers as a simple generalization of the “pick the most likely tag for this word” approach. Stochastic taggers generally resolve the ambiguity by computing the probability of a given word (or the tag). The probability is calculated using a training corpus. The training corpus is a tagged corpus, which is assumed 100% correct. The probabilities are calculated using unigram, bigram, trigram, and n-gram methods. The former method is explained in the following sections.

VI. TAG-SETS

A tag set is the set of Part of speech categories, in which, any word in a language can fall in to any one of those categories. And it gives the representation for each of the POS tag. There are various tag sets used for tagging an English corpus. The tag set for suffix stripper contains 12 major categories.

They are:
1. N - Noun
2. V - Verb
3. ADJ - Adjective
4. A - Adverb
5. Q - Quantifier
6. C - Conjunction
7. P - postposition
8. PRO - Pronoun
9. QUES - Question word
10. VBN – Verbal Noun
11. SYM - Symbol
12. NUM – Number

As the corpora envisage multiple uses, it was decided to limit the tagging only to the major twelve parts of speeches. Currently Sanskrit corpus has been tagged with more number of tag sets at word level and an elaborated labeling at morpheme level are carried out in order to meet the requirements different user group. There are 34 tags at word level and 132 tags at morpheme level.

VII. PROBLEMS PERTAINING TO TAGGING OF SANSKRIT CORPUS

1. Identification of words: Normally a sequence of characters between two successive spaces is considered as a word. It is even convenient to the computers to identify a unit as a word. But in real sense, the unit need not always be a simple word, i.e., it may be a compound or conjoined word, where the base form does not find a place in the dictionary.

2. Internal Sandhi: The morphophonemic changes that take place when a suffix is added to a stem depend on the final phoneme of the stem and the initial phoneme of the suffix and which are too many in the agglutinative languages.

3. External Sandhi: The operation of external, the morphophonemic change that takes place when two words are conjoined is not consistent in some languages like Sanskrit.

4. Inconsistency in spacing between words: In Sanskrit two or more independent words are written jointly as a single unit. Sometimes inconsistency persists in spacing between main and auxiliary verb, noun and particle, etc.

VIII. TAGGING SCHEME

The approach for grammatical tagging adopted is mainly based on the morphological analysis of these languages. To segment a word (if it has more than one morph) to its stem and suffix (es), the word can be approached either from the beginning (Left to Right) or from the end (Right to Left). The scheme, which we follow, approaches the word from the end in order to detach the suffix (es) one by one from the stem, as suffixes are finite in any natural language. The system first identifies the valid morph in the word one by one and labels them at morpheme level then the entire word is tagged for its grammatical category at word level. This system has three major components:

1. Stem- MRD (Machine Readable Dictionary)
2. Suffix – MRDs and
(3) A set of morphophonemic rules.

(1) Stem–MRD

The stem is the “main” morpheme of the word, suppling the main meaning. The major tasks involved in the preparation of stem are the collection of words, identification of their stem alternates and classification. Stem consists of all the possible roots and stems in the language.

For example, if a word has four stem alternates; the entire four stem will be included in the dictionary as independent entries. They are classified into various types on the basis of the first suffix they take. The basic structure of the stem-MRD is as follows:

| Stem / Category / Type / Status |

(2) Suffix–MRDs

Suffixes follow the stem. The basic principles underlying in the design of different MRDs for suffixes are the position of a suffix in a word and its companion. In our system the searching begins from the end of a word. The system identifies and detaches the suffix (es) one by one till it finds a stem. It is performed using a number of suffix-MRDs rather than one. The basic structure of the suffix – MRD is as follows:

| Suffix / Type / Morpheme-tag / Word-tag |

The suffix – MRD also consists of four fields. The actual suffix occupies the first field. The number in the second field indicates the type of suffixes, which could occupy the immediate left position of the present suffix. It actually helps to select the proper MRD for searching. The third field gives the grammatical information of the suffix which would be used to tag at the morphemic level. The last field indicates word-tag information, if this suffix is the determining element. The last two fields may contain more than one entry, when the suffix has different grammatical functions in different contexts. As the order of suffixation is unique for any word form, it would be easy to condition the occurrence of a given suffix. So the type number that explains this condition plays a crucial role in the analysis. S1, S2, etc. given in the type-field indicate that the possible previous element would only be a stem and that stem belongs to a particular group. The information on the stem group is made available in Sfile. The S-file for the above example is as follows:

(3) S-file

S1 > 1,2
S2 > 2
If the suffix indicates the type as S1, then the possible stems are of type 1 and 2(type given in the stem – MRD) only.

(4) Morphophonemic Rules

The third component of the system is a set of morphophonemic rules, which operate externally. It is necessary for reverting the sandhi operation in order to obtain the stem and suffixes of the word encountered, as given in the MRDs.

IX. ALGORITHM

The suffix stripper uses a list of suffixes, pronouns, adjectives and adverbs. The input format is one sentence per line in which each word is separated by a white space. On the input text, it performs the following steps:

Algorithm 1: (part of speech tagger)

POST

Step1: Begin

Step2: [initialization] Split the sentence
in to words called lexeme.

Step 3: [reading for each word]
3.1. Find the longest suffix at the End.
3.2. Find the table number of the suffix and eliminate the suffix from the word.
3.3. Go to 3.1 until the word length is 2.

Step 4: [Applying rules] Using the combination of suffixes and the rules, apply the lexical rules and assign the category.

Step 5: [Checking] For each sentence
5.1. Apply the context sensitive rules on the unknown words.
5.2. Apply the context sensitive rules on the wrongly tagged words.
5.3. If no context rule applies for any unknown words, tag it as noun.

Step 6: END.

Suffix stripper is depicted by fig 1:
X. OPERATION OF THE SYSTEM

The system reads a word from the corpus and tries to identify with those entries marked ID as status in the stem–MRD. If it possible, it finds the category in the second field and tags the word suitably. If it does not happen, it tries to categorize the last suffix and to match with suffixes, listed in L1–MRD. If it finds a match, based on the value in the type–field it proceeds to the respective suffix–MRD or the stem–MRD. This process is continued till a stem is reached. At every suitable point the grammatical value and the word–tag information (if any) is stored along with the morpheme and the position. When the word encountered is totally analyzed, the stored information as needed will be written into the output file. If the system does not find a match in the last element itself, it tries to use the morphophonemic rules to revert the sandhi operation. If there is any possibility, the system repeats the procedure from the starting. In case of uncertainty, the word will be left untagged and the next word will be taken for analysis. Similarly, and disagreement in the matching at any stage beyond L1 leaves the word untagged. As all the alternant forms of stems and suffixes are included in the MRDs, the problem of the internal sandhi is easily solved. In this model, when the system encounters more than one grammatical category for a suffix, it first attempts to analyses the whole word for the first category and then restarts the analysis for the second category, and so on. So the system is capable of analysing the homophonous forms for all their possible structures. This model also resolves the problems of compound and conjoined words which are found with
or without space to a maximum extent. Most commonly used compound forms are included in the stem – MRD. The other compound and conjoined words are tackled using repeated procedure i.e., every time after finding a stem, the system looks for any remainder. If there is, it repeats the analysis from the very beginning, as if the remainder is a new word. The untagged words and the words with more than one tag can be manually tagged.

XI. EXPERIMENTAL RESULT

One set of 100 words have been taken and manually evaluated, which gives following results. Few of them are illustrated below:

Table 1: POST output for जननी रामबन्ध्राय भोजन वचन्ति।

<table>
<thead>
<tr>
<th>Sno.</th>
<th>Word</th>
<th>Root</th>
<th>Group</th>
<th>Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>जननी</td>
<td>जननी</td>
<td>noun</td>
<td>1 relation 2</td>
</tr>
<tr>
<td>2</td>
<td>रामबन्ध्राय</td>
<td>रामबन्ध्राय</td>
<td>noun</td>
<td>2 subject 3</td>
</tr>
<tr>
<td>3</td>
<td>भोजन</td>
<td>भोजन</td>
<td>noun</td>
<td>3 object 4</td>
</tr>
<tr>
<td>4</td>
<td>वचन्ति</td>
<td>वचन्ति</td>
<td>verb</td>
<td>4 verb 4</td>
</tr>
</tbody>
</table>

Table 2: POST output for अशकः मुजेन चास परिवान्ति।

<table>
<thead>
<tr>
<th>Sno.</th>
<th>Word</th>
<th>Root</th>
<th>Group</th>
<th>Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>अशकः</td>
<td>अशकः</td>
<td>noun</td>
<td>1 relation 3</td>
</tr>
<tr>
<td>2</td>
<td>मुजेन</td>
<td>मुजेन</td>
<td>adverb</td>
<td>2 adverb 3</td>
</tr>
<tr>
<td>3</td>
<td>चास</td>
<td>चास</td>
<td>noun</td>
<td>3 object 4</td>
</tr>
<tr>
<td>4</td>
<td>परिवान्ति</td>
<td>परिवान्ति</td>
<td>verb</td>
<td>4 verb 4</td>
</tr>
</tbody>
</table>

The system gives 90% correct tags for each word.

\[
\text{Precision} = \frac{\text{No. of correctly tagged words}}{\text{No. of total words}}
\]

The sentences were taken randomly from the database and evaluated. The evaluation table is given below:

<table>
<thead>
<tr>
<th>No. of Tested Words</th>
<th>Totally Tagged words</th>
<th>Correctly Tagged Words</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>100</td>
<td>90</td>
<td>90%</td>
</tr>
<tr>
<td>100</td>
<td>100</td>
<td>91</td>
<td>91%</td>
</tr>
</tbody>
</table>

Table 3: evaluation table

The evaluation was done in two stages. Firstly by applying the lexical rules and secondly, after applying the context sensitive rules.
XII. CONCLUSION

The concept analyzed in this paper is basically evolved to handle the languages, which are morphologically rich Languages like Sanskrit. The concept is language independent. After deducting some procedures this model can be used for spell- checker as well as considering the speed, consistency, accuracy indicated by Leech (1993:279), for a tagging scheme, this system may be slow. But according to speed point of view, it need not be considered on particular with the other two criteria as tagging as corpus is a one on time job. More then that speed of application can be considerably in this concept, by building a single suffix–MRD depend on the situation only, in this case the corpora should be free from spelling and grammatical errors.

REFERENCES