International Conference on "Emerging Trends in Computer Engineering, Science and Information Technology"- 2015
Special Issue of International Journal of Electronics, Communication & Soft Computing Science and Engineering, ISSN: 2277-9477

Study of Methodologies for utilizing Sanskrit in Computational Linguistics

Prajakta R. Chaudhari **Pooja C. Gangurde** **Nikhil L. Kulkarni**

Abstract — **In this paper we are demonstrating view towards Sanskrit as a programming language. Sanskrit being considered as the purest language has the rich set of unambiguous grammar. A language to be computationally acceptable must have unambiguous grammar which is fulfilled by Sanskrit. Sanskrit has a great feature of dual case and words are represented according to properties which make it the purest among all languages. Use of Sanskrit as a computational language can avoid misconception, mispronunciation increasing the viability of language. We also provided the information about Panini's Asthadhyayi and its rule base.**

Key Words — **unambiguous grammar, Sanskrit, Ashtadhayayi**

I. INTRODUCTION

Natural Language Processing (NLP) is a wide used field for communication between computers and human language in field of Artificial Intelligence (AI) and linguistics.

NLP emerged in 1950's with attempts to automate translation between Russian and English. Modern NLP algorithms are based on especially statistical machine learning. This development provides the possibility of natural language interfaces to knowledge bases and natural language translation NLP assumes that we can teach a system about how to speak and understand.NLP is important because of its flexibility and accuracy involved with the system. Sanskrit is considered as the mother of all languages. Being one of the most ancient languages, Sanskrit has the strongest and simplest grammar of all the natural language and also the most suited for Artificial Intelligence and NLP. It is commendable that Panini was able to design a language that can make computers understand the concept of human linguistics without any ambiguity. Considering the semantics Sanskrit immensely useful for creating a highly efficient Natural Language Processor.

II. FEATURES OF LANGUAGE TO BECOME COMPUTATIONALLY VIABLE

A. *Less or Unambiguous Grammar*

The most important tool for communication is language. To express our views towards something we require language. A well known language is easy to understand, as in case of communication if the persons communicating are well known to a single language then the process is much easier, than that if both of them well known to different languages [1]. A computationally viable language is a language used in field of computer science for interaction within the system. Grammar is the most important aspect of any language. The syntax and semantics are the major aspects of a language. The language to be used as a computational language must have a rich set of unambiguous grammar.

Ambiguous grammar is a context free grammar having more than one derivation (i.e. left and right), as a result of which complications arise. So a computational language must be unambiguous having only one derivation (i.e. left) avoiding complications.

B. *Mispronunciation/ Misspelling Resulting in Misconception*

As we know that a language to be computationally viable must have unambiguous grammar. Many of the languages are ambiguous because of mispronunciation and misspelling of the words [1].

Let us consider the example of English:

- A good life depends on a **liver**

By observing this sentence the word **liver** has two meanings, **liver** can be the organ in human body and it can also be a living person.

This sentence may cause misconception as they have same pronunciation and spelling.

- Also the words **hole** and **whole** have same pronunciation but totally different meaning.
- Also, DEER and DEAR:
  DEER-name of animal
  DEAR-it means lovable

Thus a computational language must guard against such ambiguity.

C. *Total precision*

Precision means the quality or condition to be accurate. Total precision ensures that the computational language must perform its task accurately and precisely without changing its form, which means it should be grammatically correct.

A language having accurate grammar designed according to its rule base can be considered as a computational language.

III. SANSKRIT AND IT'S FEATURES

A. *Unambiguous grammar*

Sanskrit is considered as the mother of all languages. It being the most ancient language was called as "DEVVANI" (language of god)[3]. It is the purest among all languages as a reason of which many word from different languages are derived from Sanskrit. Like for English मातृ becomes Mother, भ्रातृ becomes brother. Sanskrit grammar was designed by Maharshi Panini around 2500 years ago it is has unambiguous grammar [3].

As compared to other languages available Sanskrit grammar is very vast and intelligent with a guard against ambiguity.

A. *Special attention to dual case*

Dual case is an interesting feature in Sanskrit. This feature makes it a unique language. The representation of singular case, dual case and plural case is a problem in most of the languages which is avoided in Sanskrit.

Mainly the problem is in dual and plural case like for English between 2 **boys**(dual case) and among many **boys**(plural case), same as for Spanish entre los **chicos**(dual case) and entre los **chicos**(plural case). The same in Sanskrit is बाल्कयो-means between 2 boys (dual case) and बालकेषु-means among many boys (plural case) [3].

A comparative study between English and Sanskrit demonstrating Dual case:

- ENGLISH:

-Year (singular) -2 **years**(dual) -3 **years**(plural)

Dual and all plural forms have same word **years** which may led to confusion.

- SANSKRIT:

-Year (वर्ष) -2 Years (द्विवर्ष) -3 years(त्रिवर्ष)

In Sanskrit each case is represented using different words thus eliminating ambiguity.

B. *Word representation based on property instead of Objects*

In any language words are given names (word representation) according to the object it contains or the object itself due to which the words are flooding in the dictionary. The word representation in Sanskrit is not done according to the objects; instead it is done by its property. Any object or a thing is named by the property it possess like पदप्- means property of drinking water using foot (tree). अमृत- means not dead, here मृत is dead and अमृत means not dead [3]. As the words are represented by its property each one can name it accordingly and no need to derive its name and add in dictionary for others knowledge. Each person can understand any new word easily as it specifies its property only.

C. *Co-relation between written and Spoken form of Words*

We have seen many times that some words are written in a way and spelled other way, like **psychology** in this word **'p'** is written but not pronounced. This concept is eliminated in Sanskrit. Sanskrit has same written and spoken word having one-to-one correspondence.

Example:

मातृ, देववाणी, संस्कृत etc.

D. *Potential Grammatical Tools*

Sentence formation in Sanskrit is done with the help of two well known tools Vibhakti and Karaka. Vibhakti guides for making sentence in Sanskrit, there are seven kinds of vibhakti and they also provide information on respective karaka. These seven vibhkti's are :

- Prathama –Nominative
- Dvitiya –Accusative
- Tritiya –Instrumental
- Chaturthi –Dative
- PA.Nchami –Ablative
- Shhashhthi –Possessive
- saptami –Locative
- Sambodhana –Denominative

Karaka approach guides for generating grammatical relationship of nouns and pronouns for other words in a sentence [4].

IV. POTENTIAL GRAMMAR OF SANSKRIT, USE OF NLP AND NLU

A. *Ashtadhayayi*

How a language communicates information?

This question led to different theories for language analysis, Paninis grammar saw different thoughts in his work Ashtadhayayi. Ashtadhayayi is written to describe the most powerful Sanskrit language. Ashtadhayayi consists of around **4000 rules** with some ancillary texts.

- sivasutras (the special order of the phonemes)
- dhatupatha (list of a verbal roots)
- ganapatha (the various sets of nouns)
- linganusasanam (system that deciding the gender)
- unadisutras (some special rules).

Study of Ashtadhayayi from computational point of view may lead to a new programming paradigm as it has rich structure [2].

B. Rule base of Sanskrit

Sanskrit rule base consists of **3949 rules**. Some of the sutras in Sanskrit are:

Samjna Sutra:

It assigns attributes to the input string thereby creating an environment for certain sutras to get triggered [4].

Adhikara Sutras:

It assigns necessary condition to the sutras for getting triggered (χ) [4].

Paribhasha Sutras:

It takes decision and helps us in resolving conflicts and deadlock conditions. It also provides a meta language for interpreting other sutras [4].

Some other points which makes Sanskrit grammar as a potential grammar:

- Distinguishing mechanism provision.
- Generative as well as descriptive.
- Provides exhaustive grammar results into efficient language analysis.

V. USE OF NLP AND NLU IN SANSKRIT

A. NLP Framework

NLP is used for communication between computers and human language in the field of Artificial intelligence and linguistics. Sanskrit as having rich set of unambiguous grammar is used for NLP.

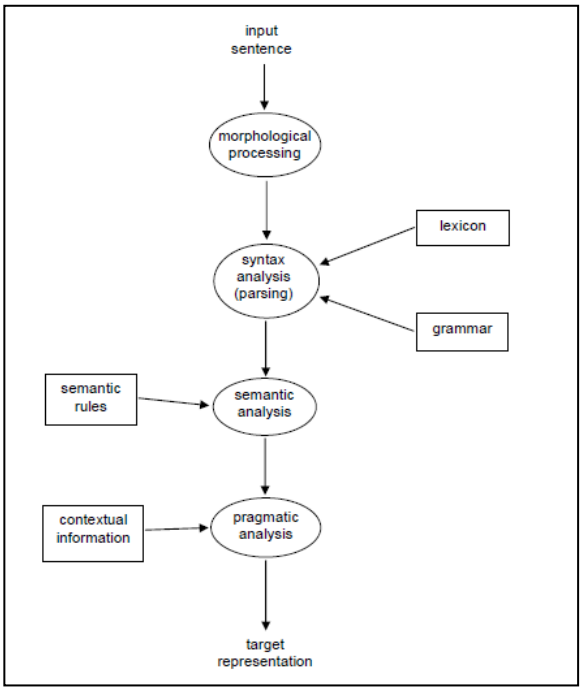

Fig.1. NLP Framework

B. Unambiguous Grammar

Comparative study for avoiding ambiguity in Sanskrit:

- Example:

Let us consider the given English sentence –"I like apple"

1. Does this sentence refer to the brand apple, or to the fruit?

2. Who is "I" in this sentence? [3]

- Example:

Let us consider the given English sentence –"Do you see the man with the glasses?"

1. Does that mean -"you see the man using the glasses"

2. Or does it mean -"you see the man who is holding the glasses"

3. Glass here refers to spectacles or normal glass? [3]

Here grammatically both sentences are correct, but which meaning was implied here, that depends on the present context. Thus NLP has to face a lot of ambiguity related problems during its processing.

- Now form following example we see that how Sanskrit language overcomes all those hurdles to become suitable for NLP [3].

मूर्खः परिहर्तव्यः प्रत्यक्षः द्विपदः पशुः ।

मूर्खः ⇨ Stupid

परिहर्तव्यः ⇨ Avoidable

प्रत्यक्षः ⇨ In front of eyes

द्विपदः ⇨ Having two legs

पशुः ⇨ Animal

The above example illustrate that each word in Sanskrit is represented according to its property. In Sanskrit there is no need to define a word for each object or no need to surf the dictionary for meaning, the meaning of the word or object is given according to the property of the object.

Like in the above example **property of being stupid**, **being avoidable**, **being in front of eyes**, **property of having two legs** is given, which clearly state the meaning of each word and the whole sentence.

This sentence means- **"A stupid person must be avoided. He is like a two legged animal in front of the eyes."**

VI. HOW SANSKRIT IS USED IN OOP

OOP is incapable in designing and simulation of expert systems with their current language constructs.

Some features of Sanskrit like:

- No Punctuations
- Word representation based on properties
- Special attention to dual Case
- Co-relation between written and spoken form of words

Helps us to consider it as a candidate for computer programming in fields of Natural Language Processing(NLP) and Artificial Intelligence(AI) [1].

CONCLUSION

Thus study of Sanskrit grammar with respect to Ashthadhayayi and studying the rule base of it gives a bright view towards considering Sanskrit as computationally viable. The great features of Sanskrit can be used to develop a application and also a view towards designing dependency parser.

REFERENCES

[1] P. Ramanujan "A Case for Sanskrit as Computer Programming Language" .
[2] Akshar Bharti, Amba Kulkarni,Department of Sanskrit Stuies University of Hyderabad"Sanskrit and Computitional Linguistics" 30th Oct 2007.
[3] Deeptanshu Jha, Dr.Rashmi Jha, Varun Varshney "Natural Language Processing and Sanskrit"
[4] Shashank Saxena, Raghav Agrawal " Sanskrit as a Programming Language and Natural Language Processing" C.S C.S, IIET IIET

AUTHOR'S PROFILE

Prajakta R. Chaudhari
UG Scholar
Sandip Institute of Technology and Research Centre, Nashik
Savitribai Phule Pune University, Pune
praju7994@gmail.com
Intrested in NLP, AI.

Pooja C. Gangurde
UG Scholar
Sandip Institute of Technology and Research Centre, Nashik
Savitribai Phule Pune University, Pune
gangurde26pooja@gmail.com
Intrested in NLU, Expert System.

Nikhil L. Kulkarni
Assistant Professor
Sandip Institute of Technology and Research Centre, Nashik
Savitribai Phule Pune University, Pune
nikhil.kulkarni@sitrc.org
Intrested in Compilers