A Review of English to Indian Language Translator: Anusaaraka

Kanika
AIM & ACT
Banasthali Vidyapith
Rajasthan, India

Ankur
AIM & ACT
Banasthali Vidyapith
Rajasthan, India

Divyanjali
AIM & ACT
Banasthali Vidyapith
Rajasthan, India

Shalini Mittal
AIM & ACT
Banasthali Vidyapith,
Rajasthan, India

ABSTRACT

In this paper, we present the concept of Anusaaraka. Starting with a small introduction of natural language processing to understand the entire concept we move on to detailed overview of Anusaaraka. India is very rich when it comes to languages hence entirely different rules are needed to be incorporated for each language. Indian languages are free-word order languages hence should be dealt accordingly [16]. Anusaaraka is a kind of language translator. It is designed to translate English to any Indian language. Outlines of some other Indian language translators developed during recent years are also given in short to provide a overview of the progress in the field of translation. The key features of each one are presented and analyzed to meet the requirements of an ideal system.

General Terms

Natural language processing, Anusaaraka, Indian languages, Sampark

Keywords

Anusaaraka, Local word grouping, Free word order languages, Paninian model

1. INTRODUCTION

Dealing with natural language and processing it has always been a significant issue in the field of artificial intelligence and man machine interaction [12]. Human mind encodes the feeling or message in his language and pass it on to the receiver. While encoding the speaker has a model of the listener regarding the knowledge he has. Based on that he may not include some information that he assumes the listener knows [1]. The other person after receiving the message decodes it to understand what the speaker has to say. This can be said as processing the language. Decoding of the sentence involves two kind of knowledge. First is the language knowledge such as the rules used while encoding, grammar, lexicons and some other features of the language. Second is the background knowledge which includes domain specific knowledge and context etc to understand the meaning. Natural language processing aims at developing computational models that can read, understand and analyze the language by making these two types of information available to the system itself. This model deals with operations like sentiment analysis, parsing, extracting relationships between sentences and many other such functions in order to generalize the sentence structure and derive its meaning correctly. In general sense natural language processing tries to make the computer understand what a human wants to convey in his/her language. Due to this role it is more involved in human computer interaction.

Some of the main applications of understanding the languages are

- Information extraction
- Text summarization
- Man machine interaction
- Expert systems

A lot of work is being done in the field of natural language processing from the beginning of the time. But the outputs were not error free in any of the attempt.

Recently it was observed that if the system developed deals with entire discourse not the sentences individually, then the results will be more appropriate and no ambiguity will be there [4]. Document translator mantra has been designed by keeping in mind such a strategy only. It works on lexical tree to lexical tree translation not on word to word translation [25]. Following subsection describes the stages involved in processing the natural language.

1.1 Morphological Processing

As soon as any sentence is given as input to the system its morphological processing starts. Morphological processing is preliminary stage that preprocesses the sentence before passing it on to the syntactic analyzer [8]. The sentence is fragmented into set of tokens corresponding to unique and defined words, sub words or punctuations. The input was in the form of a string and after this stage it is converted into discrete sets of tokens and passed on to the next phase. There are two ways for accessing the meaning of the word. One is that you directly access it as we do it while referring it in dictionary. Another is the indirect method where you get the word by reaching to it through the morphemes. The first case is full parsing and the indirect one is called full listing [2]. Morphological information is not utilized in case of full listing models. These models use associative and rapid procedures and consider that at the stage of access. Hence it is said that full listing method requires some pre-lexical treatment of morphological constituents. There is one more model that is generated by combining both the previously stated models called dual or mixed model.

While assigning tokens we need to consider all type of words. Base words can be easily identified but words may appear in modified forms also. Modification can be done by adding prefixes or postfixes. In these cases it is simple to extract the base word but sometimes words can be modified due to inflections, derivations or compounding [3]. Depending on the language being processed the morphological analysis may vary in its processing. English can be analyzed orphologically with relatively less ambiguity and easily than many others.

The output of morphological analysis stage is collections of tokens instead of words. These tokens are further processed by the next stage. There are two kind of approaches used: language dependent and language independent approach base on which morphological analyzer can be designed [14]. Many morphological analyzers are developed that can successfully do the morphing for a variety of languages starting from Arabic to Chinese, English, and European languages using the available approaches [13]. In 1983 a researcher developed two level morphological analyzer with the first level being the description of word in the order they appear and the second being the lexical level [19].

1.2 Syntax Analysis

It is the formal processing by a computer to break the sentence into small clusters to clarify the meaning more clearly and less ambiguously. There are two operations that are performed during the syntactic analysis. The very first operation is to assess whether the words in the sentence are well formed and appropriate to analyze or not. Second major operation is to divide the entire sentence into group of words that are syntactically related. The words are grouped into one set if they show any kind of relationship with other words in that group. Syntax analyzer is sometimes referred to as parser. Its analysis is then called parsing.

Every language convertor needs to incorporate parsing as its integral part. There are two components that helps the parser to accomplish its task. These are grammar and lexicon [3]. Lexicon denotes the category to which each word syntactically belongs and grammar is collection of rules that are followed while making the sentence. Every language has its own rules or in other words every language has its own and unique grammar. But in real world problems there are a lot of ungrammatical sentences that do not follow the rules. Any system that looks forward to understand language must also be able to describe these utterances.

1.3 Semantic Analysis

Semantic structures are more helpful in moving towards translation than syntactic one due to two reasons. First is that semantic roles tend to agree better between two languages and second is the set of semantic roles of a predicate models the skeleton of a sentence, which is crucial to the readability of MT output [6]. Semantic analysis is the study of semantics, meaning of speech and how the sentence is structured. This study aims at finding out the meaning of conversational speech, detect grammatical patterns, and to discover specific meaning of words in a particular language. In simple words semantic analysis is the process of understanding the verbal communication, be it formally used to convey a message or informally uttered. Wittgenstein described understanding as knowing how to use it. For understanding we need to interpret and derive its meaning. Semantic analysis relates the words to their corresponding language independent meaning. The language specific features are removed from the words during semantic processing to whatever extent it is possible. The lexicon discussed in previous subsection must be elaborated to have semantic definitions and the grammar also must be made to incorporate rules to specify how the semantics of any phrase are formed from the semantics of its component parts. This stage identifies each word with its general meaning that is not language specific but general to all the languages. It figures out the meaning of the linguistic input.

There are some commonly used approaches that the semantic analyzers use. These are

- Statistical approach
- Information retrieval
- Domain knowledge driven analysis

Lexical semantics deals with meaning of component words and sometimes also handles word sense disambiguation [7]. Word sense disambiguation resolves the ambiguity of one word may make different senses in different environment. Compositional semantics deals with how words combine to form different compound words. Some people prefer to address all these issues in pragmatic analysis as a different stage which is discussed in later section. At times it is considered that semantic analysis also needs syntactic analysis and pragmatics by its side to accomplish its task but in this paper we have described it as a separate stage in natural language processing. Semantic analysis is a bit difficult than it appears to be. This difficulty is there due to ambiguity in languages, common sense knowledge requirement and dynamic nature of language.

1.4 Pragmatic Analysis

Context of the conversation or sentence has an important role in determining its correct sense. Hence other than the grammar and lexicon categorizations it is important to identify what is the perspective of the conversation. Same words may have different meanings depending on the background. Pragmatic analysis deals with that context of the sentence only [5]. The context is significant as the meaning of the entire conversation can be different based on not only the words used and the structure or rules followed but also on the circumstances in which it is going on. Ambiguity may arise in understanding the meaning of words said in one context and same words said in other situation.

The issue of pragmatics is considered to be fuzzy. The results of semantic analysis are processed by the pragmatic analyzer. Some language processors do not necessarily differentiate between semantic and pragmatic analysis but Russel and Norvig states that pragmatic analysis determine the context of the sentence after semantic analyzer determines its meaning. The entire interference can be drawn only with the help of pragmatic processing. Semantic analyzer work on word level but pragmatic analysis is carried out by taking into account the discourse. By looking at the single sentence sometimes it is not possible to clearly reach at the background of the conversation.

Semantic analysis is entirely based on the semantic rules determined by the language and system. Pragmatic analysis is based on the contextual information fed into the system [24]. If more than 1 sentence creates ambiguity then by using pragmatic analysis one out of the options made available by the semantic analyzer can be chosen.

The final output is generated after passing the sentences through the above stated four basic stages. Figure 1 is the flow diagram of a natural language processing system. It depicts the sequence in which processing occurs and the output is generated.

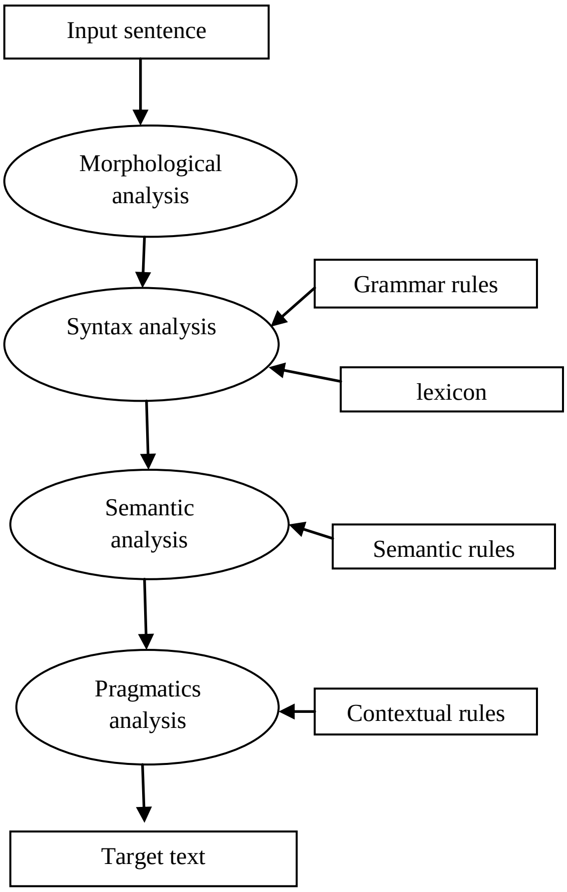

Fig 1: Flow chart of Natural Language Processing

2. LANGUAGE TRANSLATORS

Language translation started with an effort of translating almost 60 Russian sentences into English during an experiment called Georgetown experiment which was carried out in mid 1950's. At that time researchers took it as an easy problem and it was forecasted that machine translation would be solved completely soon. But later on the progress was not so fast. Till today efforts are being put to develop a translation system that can translate any text from source language to target language with high precision and no limitations. For the duration of 15 year starting from 1964 the dark period with no progress in the field of machine translation passed on. During this period it was stipulated that machine translation is not at all possible in future which was just the opposite of what was expected in past. It suddenly started reviving after 1980's. Then in 1990 a new era took its height for language processing or machine translation. In 1996, Beesley developed a finite state transducer of Arabic language for MA by applying XFST. This transducer uses xerox finite state transducer by reworking expansively on the lexicon and regulations in the Kimmo-style [15].

The results of these continuous efforts is that the systems are improving but at a very slow pace. The main hurdles in the course of developing a perfect translator are some of the language features themselves. These features are ambiguity as it is listed before also. Incomplete and ungrammatical sentences are also a major problem while translating and referring them in the database. The way a human encodes the sentence also creates problem.

Despite of these difficulties there are systems that produces correct output but with strict constraints. Some of the translators developed in India are listed below with their key features.

2.1 Mantra

Mantra is a MAchiNe assisted TRAnslation tool. It was developed by C-DAC and is used by government of India to translate documents [10]. It is a domain specific tool that operates under certain domains. It deals with documents related to administration, information technology, agriculture, health care and some more. Initially it was capable of converting English to Hindi but now other Indian languages are also being dealt with. Mantra incorporates TAG (Tree Adjoining Grammar) for parsing purpose [18]. The key feature of mantra is that the format of output can be user specified. It is of great benefit as we use this translator for formal document translations. Another advantage of using mantra is that it accepts input in more than one format. Many pre-processing and selection tools are also available to refine the output [10].

One more domain specific translator is available that is implemented for tourism and health domains. It is named Anuvadaksh and can translate English to Hindi, Marathi, Bangla, Oriya, Tamil and Urdu [21].

2.2 Angla Bharti

Angla Bharti is translator for Indian languages. It uses pattern directed approach along with using structures which resembles context free grammar. It evaluates the source language, which is English here, and produces an intermediate code. This code/structure is called Pseudo Lingua for Indian Languages (PLIL). This structure in the next step is transformed into the destination language. By using a process of text generation the intermediate code is brought into the frame of Indian languages [11]. With this system there is a requirement for automatic pre-editing the sentence, paraphrasing, and identification of named entities. Two modules are also incorporated; these are statistical language module and error analysis module for automated post editing. The aim of pre-editing is to convert the input sentence into a form which is more easily and correctly translatable.

2.3 Shiva

It is a machine translator that follows a corpus based approach for converting the source text into target text. In corpus based approach the system tries to learn the necessary rules for translation from a parallel corpus [23]. Shiva works on the primitive word to word translation. It is an example based system that requires very large parallel corpora. It is observed that corpus based approach is time consuming [8].

2.4 Shakti

It works by combining the two different approaches. Combination of rule based and corpus based approach governs the translation process. Along with the rules defined by the language it also incorporates the knowledge gained by corpora while translating. Like the other translators it is also working on Indian languages and English.

2.5 UNL based Translator

UNL stands for universal networking language. It is a formal language that is specifically used to represent semantic data extracted from natural language texts. It was developed by a professor of IIT Mumbai. UNL based translator uses UNL formalism to translate English to Bengali and Marathi language.

2.6 Sampark

It is developed by consortium of institutions including IIIT Hyderabad, IIT Kharagpur, CDAC and some more in the year 2009. It can translate Punjabi, Tamil, Telugu, Urdu and Marathi to Hindi. Also it can convert Hindi to Punjabi, Tamil and Telugu inter-conversion till now.

3. ANUSAARAKA

The name Anusaaraka is derived from Sanskrit word Anusaaran that means "to follow". In the processing of Anusaaraka output appears in one step followed by the next one. Hence it is named so based on its way of generating the output. Anusaaraka is a translator that accepts English as input and produces output in Hindi. The sentence is passed through various stages of defragmentation and analysis before the output is generated.

Encoding the thoughts is done by human beings and the output is sentence. It is string of words and words are sequence of characters separated by punctuation marks or spaces. India is a large country with diversity in terms of languages and most of them are free word order languages and for these kind of languages paninian grammar is best suited [20]. Paninian framework was designed for writing Sanskrit grammar decades ago. But as all the Indian languages have something in common with it, paninian framework can be made as a base for the parsers. There are two min parts of the parsers. Lexicon is language dependent but parser does not depend on language. Hence it is a great benefit that the same parser can be used for almost all Indian languages just by changing the lexicon. It is therefore adapted to deal with many Indian languages. It uses two kind of information: one is vibhakti knowledge that is primary deciding parameter for mapping of semantic relationships and other one is position information that is the secondary parameter [1].

Anusaaraka translates English or any other local language to Hindi and other Indian languages. The translation output is presented to the users in layered form starting from source to target language. It is the only translator that aims at bridging the gap between two languages by producing transparent results step by step, thus fulfilling its name. There is a difference between working of Anusaaraka and other language processors. Anusaaraka works towards first understanding the source language. Reading and getting the exact meaning is the main concern rather than just translating and not worrying about the correct portraying of the meaning of the sentence. The layered output significantly controls the flow of information in a particular manner an one can refer to the previous stage without loss of information. Transparency and reversibility are the two key features that differentiate Anusaaraka from other translators. The principles followed by developers of Anusaaraka are that there is never a loss in doing an effort even if it is not successful and the competence is developed out of willingness.

Here we mainly discuss how Anusaaraka deals with Sanskrit and English. The next section describes the framework used by the translator. As described earlier also that paninian framework is the most general one used while dealing with Indian languages so we explain its structure first and then move on to the other components.

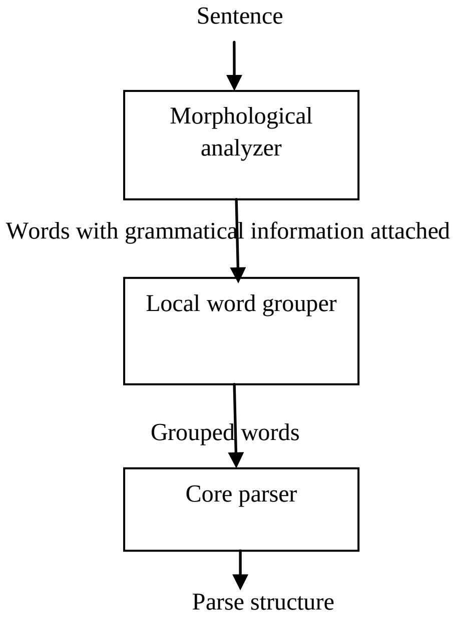

Fig 2: Structure of the Parser

The morphological analyzer takes the entire sentence as input and retrieves all the information related to it. The information is regarding its tense lexical category, gender etc.. If there are more than one meaning associated with a word then all the information regarding each word is returned from the database.

Local word grouper categorizes the words based on some local information [1]. Local information is the words that surround that particular word. Local word grouper works differently from morphological analyzer. Only those words are placed in the same group which clearly belongs to that. In case of minor ambiguity also a separate group is formed for the word rather than placing it in any of the existing groups. This block reduces the pressure of the core parser and increases the systems efficiency. The output of local word grouper is passed to the core parser which then generates the parse structure.

When we look at Anusaaraka, during its translation karaka relations between verbs and nouns are identified. It is based on the concept of demand and merit. Some words individually or in group make demands and other satisfies it. The key theme of the core parser of Anusaaraka is "aakaankshaa" and "yogyataa". Aakaankshaa here stands for demand and Yogyataa is the ability to fulfill that demand. This ability is not present in all the words. Only a few nouns with desired parsarg and semantic properties have that eligibility [1]. This is how parsing of any sentence is done by the building blocks of paninian parser.

3.1 Levels in Paninian Model

There are four level described in paninian model.

- **Semantic**: This level shows what exactly the speaker has in his mind. What the speaker wants to portray. This is the final meaning level.
- **Karka:** This level holds within it the 6 type of karaka relations and some other relations like

purpose. It has relationship with both the syntactic as well as semantic level.

- **Vibhakti:** This level abstracts away from many minor differences among languages like idiosyncratic and orthographic differences**.**
- **Surface:** It is the uttered sentence, the basic level of the model.

Group of researchers that were working on Anusaaraka have also been working for morphological analyzer for Tamil [17].

3.2 Structure of Anusaaraka

A sentence first enters the morphological analyzer which finds each word in the dictionary of indeclinable words and returns its grammatical features. If the word is not found then morphing refers to word paradigms to find whether it is possible to derive the word from root and its paradigm. if it cannot be derived then its passed to the sandhi package as it may be a compound word and analyzed again. The output of morphological analyzer is passed to local word grouper which groups words based on the local information available. After grouping sentential analysis can be done if a large database is available [1].

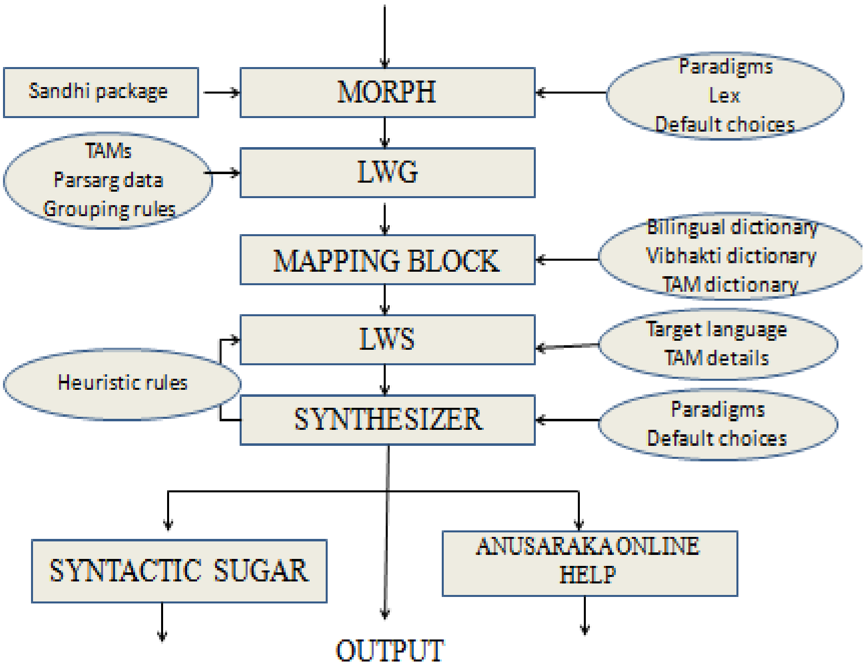

Fig 3: Block diagram of Anusaaraka

In the next stage using various dictionaries, Anusaaraka finds root and vibhakti for each word in target language. This is the first step in translation. Before this mapping stage the system was trying to understand the meaning of the uttered sentence. [22]. The word groups formed by the local word grouper are now split back by the local word splitter. In the last stage the synthesizer takes the output of splitter and generates words from root and grammatical features.

4. CONCLUSION

It can be seen that a lot of efforts are being put in the field of natural language processing towards building a faultless system with no constraints and limitations as it is not there while carrying out any conversation. We have seen some key features of the systems developed and used nowadays. All of them work on Indian languages. Some are only capable to deal with English and Indian languages but some work on Indian language pairs also. Anusaaraka focuses on access of Indian languages only. Although it is not implemented on a large scale but it aims at perfect information preservation. It can be considered best for Indian languages because all these languages are free word order languages. The unique feature that makes Anusaaraka different from other machine translators is its transparency in computing the output. The output is visible at each layer on the screen. It is reversible in its working also as we can traverse back to the previous stage without the loss of any information that may not be there in any other translator. Its central aim is to understand the implication accurately and deliver it to the user. It can successfully bridge gap between two languages with some improvements. Work is still going on in the field of natural language processing as well as Anusaaraka is also not fully developed yet. Hence following its principle of any effort is never harmful even if it does not provide success we should try to remove the shortcomings.

5. ACKNOWLEDGMENTS

We thank Banasthali University for providing us resources due to which we were able to accomplish study on natural language processing. We are really grateful for the kind of environment and motivation which we have received from the institute. Without their help it would not have been possible to work in this direction all alone.

6. REFERENCES

[1] Akshar Bharti, Vineet Chaitanya and Rajeev Sangal, 1994 Natural Language Processing: A Panini Perspective, Prentice Hall of India.

[2] Alberto Dominguez, Fernando Cuetos and Juan Segui, 2000. Morphological processing in word recognition: A review with particular reference to Spanish data, Psicologica, International journal on methodology and and experimental phsycology, volume 21, 375-401.

[3] Natural language processing, chapter 2, Lkit: a toolkit for natural language interface construction.

[4] Peter Norvig and Stuart J. Russell, 1994, Artificial Intelligence: A Modern Approach, 3rd edition.

[5] Levinson, Stephen C. (1983) Pragmatics. Cambridge University Press.

[6] Ding Liu and Daniel Gildea, 2010. Proceedings of the 23rd International Conference on Computational Linguistics (COLING-10), Beijing.

[7] Vera Aleksić & Gregor Thurmair, 2011: Personal Translator at WMT 2011 – a rule-based MT system with hybrid components. [WMT 2011] Proceedings of the 6th Workshop onStatistical Machine Translation, Edinburgh, Scotland, UK, July 30-31, 2011; pp.303-308.

[8] Antony P.J. and Dr. Sonam K.P., 2012. Computational Morphology and Natural Language Parsing for Indian Languages: A Literature Survey, International Journal on Scientific and Engineering Research, volume 3, issue 3.

[9] Chaudhury, S., Rao, A. ; Sharma, D.M., 2010. Anusaaraka: An expert system based machine translation system, International Conference on Natural Language Processing and Knowledge Engineering, 1-6.

[10] http://pune.cdac.in/html/aai/mantra.aspx

[11] http://tdil.mit.gov.in/research_effort.aspx

[12] Patten T., Jacobs P., 1994. Natural Language Processing, volume 9, issue 1.

[13] Shambhavi. B. R, Dr. Ramakanth Kumar P, Srividya K, Jyothi B J, Spoorti Kundargi, and Varsha Shastri G, 2011. Kannada Morphological Analyser and Generator Using Trie, International Journal of Computer Science and Network Security (IJCSNS), VOL.11 No.1

[14] Choudhary and Narayan Kumar, 2006. A computational framework for the verb morphology of great Andamanese, JNU.

[15] K. Beesley, and L. Karttunen, 2003. Finite State Morphology, CA: CSLI Publications, Stanford.

[16] Akshar Bharati and Rajeev Sangal, Parsing free word order languages in the Paninian framework, www.ldc.upenn.edu/acl/P/P93/P93-1015.pdf.

[17] S. Rajendran, “Parsing in Tamil”, Language in India, www.languageinindia.com Volume 6 : 8 August, 2006.

[18] Akshar Bharati, M Gupta, Vineet Yadav, Karthik Gali, and Dipti M. Sharma, 2009. Simple Parser for Indian Languages in a Dependency Framework, In Proceedings of the Third Linguistic Annotation Workshop (LAWIII), SIGANN, 47th ACL - 4th IJCNLP, Singapore.

[19] Uma Parameshwari Rao G, Parameshwari K: CALTS, University of Hyderabad, On the description of morphological data for morphological analyzers and generators: A case of Telugu, Tamil and Kannada.

[20] Sobha Lalitha Devi and Menaka S, May 2011. Semantic Representation of Causality, Special Volume: Problems of Parsing in Indian Languages.

[21] Bharati, Akshar, et.al, 2002. Anusaaraka: Overcoming the Language Barrier in India, appeared in ”Anuvad”, Sage Publishers, New Delhi.

[22] Amba P. Kulkarni, 2003. Design and Architecture of ‘Anusaaraka’- An Approach to Machine Translation Satyam Techical Review, vol 3.

[23] Gregor Thurmair. 2004. Using corpus information to improve MT quality. In Yuste Rodrigo, Elia (ed) Paris: ELRA (European Language Resources Association): Proceedings of the Third International Workshop on Language Resources for Translation Work, Reseach & Training (LR4Trans-III).

[24] Gregor Thurmair, 2009. Comparing different architectures of hybrid Machine Translation systems. Proceedings of the Twelft Machine Translation Summit. Ottawa, Canada. 340-348.

[25] Galley, Michel, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What’s in a translation rule? In Proceedings of NAACL-04, pages 273–280.