"Sanskrit: Some Insights as a Computer Programming Language"

Kinjal V Patel

Research Scholar *in the Sanskrit Department*

Department of Mathematics, Hemchandracharya North Gujarat University, Patan, Gujarat, India

Aims to explore how the Sanskrit Language can be used as a Natural Processing Language.

Abstract: **Now a days it is in the "Sanskrit World", heard the voice that "Sanskrit is the best language for computer programming". Being some familiar with computer programming, I wondered if there was any truth to that statement. The claim is based on this paper by Rick Briggs, published in AI Magazine [1]. The paper talks about using Sanskrit in natural language processing (NLP). The idea of using a natural language for computer programming is to make it easier for people to talk to computers in their native tongue and spare them the pain of learning a computer friendly language like assembly/C/Java. In this paper we exhibited all the environments related to Sanskrit Language, Computer Programming Language and Natural Language, Processing so that the Sanskrit Scholar can be categorize the promises of the language suitable for NLP as an AI.**

Keywords: **Sanskrit, Computer, Natural Processing Language, Programming Language, AI- Artificial Intelligence.**

I.   INTRODUCTION

There was a paper by Rick Briggs, a NASA researcher, published in the spring issue of Artificial Intelligence magazine in 1985 (Volume 6 Number 1), entitled 'Knowledge Representation in Sanskrit and Artificial Intelligence'. It can be found here on AAAI's website [1]. The Rick Briggs' paper makes a case that natural languages are not that difficult to use for computer programming. He cites Sanskrit as an example as its grammar can be easily translated to a form understandable by a computer. But nowhere does it say that Sanskrit is the best way to program a computer. Hence we say, if somebody manages to create a computer language based on Sanskrit, then the question arises that: How likely is it that it will get adopted by non-Sanskrit speaking people?

I spend a lot of time on the internet searching for interesting things to learn for my research topic: "e-Resources of Sanskrit". The search isn't specific and neither is the inflow of information. I happened upon a few articles recently that suggest Sanskrit, the ancient nigh dead Indian language, is good for computer programming and that NASA uses it to program artificial intelligence[2][3][4]. A peek at the headlines triggered my bullshit alarm – we should all have one – but, in those moments of curiosity, I perused their contents. They were so very devoid of rationality, I had to search for a fun little activity to take my mind off it. Betteridge's law [5],[6] of headlines did the trick. The **Betteridge's law state that "**If the headline to an article is a question, the answer is always no." Also on the internet, I found the best comments for Rick Briggs is that: 'Rick Briggs' at best is a pseudonym [5]. There are absolutely no other works in related fields attributed to this name.

It begins with Briggs describing the current state of events surrounding artificial intelligence. It had been quite an undertaking to design unambiguous representations of natural languages for the purpose of computer processing. Natural languages – the way humans communicate with each other – had not been easy to parse and transform into information that a machine could understand. Even if they could overcome that barrier, there was the issue of ambiguity – statements could mean different unrelated things, depending on the context. A human who spoke the language wouldn't find it hard to understand what was actually meant, but computers would. It led to the belief that there might not be a way to effectively exchange information with machines without the help of an artificial language [7].

Briggs, in his paper, challenged that belief by drawing attention to the fact that there has existed at least one natural language which could, in theory, be used as an artificial language. It had a logical structure that mapped on to certain knowledge representation schemes perfectly. That language, of course, was Sanskrit.

II.   THE CONTENTS OF THE PAPER.

The paper provides a whole lot of compelling arguments that show it is indeed possible for a natural language to work as an artificial one. That's it. It does not at all claim that Sanskrit has to be that language. Sure, it uses Sanskrit as a case study, but that's all there is to it. A quick read of the 8 page piece should make that clear. Here's a gist of the points he tried to make. I've used my natural intelligence to summarize it. It's not chronological but exhaustive.

A. A perfect natural language must have these characteristics.
- A statement should be easy to break down into a semantic net or an array of semantic data

[8],[9],[10].(He referred to the array as series of triplets.)

- It should be easy to compile a natural language statement from the data array. It should be human readable and understandable.
- The statements coming out should be about the same as the ones going in. It shouldn't sound weird, nor should it loss or gain information.
- Deviations if any should be minimum.

B. Sanskrit, as it turns out, does all of that. It has an extremely logical structure. Its grammar rules allow a kind of precision unmatched by other languages. It has a near unchanging syntax.

C. The computer readable data representation of a Sanskrit statement can be obtained by simply placing the individual words of the sentence in an array. This is aided by the fact that word order simply makes no difference in Sanskrit.

D. That very sentence can be reconstructed by putting together the contents of the array.

E. The language is extremely concise. It has perhaps the highest information to word count ratio [11]. There are no redundancies.

Besides above, we also find the following from the last paragraph of Briggs' paper.

It is interesting to speculate as to why the Indians found it worthwhile to pursue studies into unambiguous coding of natural language into semantic elements [12]. It is tempting to think of them as computer scientists without the hardware, but a possible explanation is that a search for clear, unambiguous understanding is inherent in the human being.

Here note that the conclusion of the paper was that humans are capable of using an extremely precise unambiguous language. That should save you some back and forth when we debunk baseless claims.

III. SOME TRUTHS ABOUT SANSKRIT:

Sanskrit is a brilliant language. I'm not kidding and neither is I being ironic. It really is the most precise language in existence, with Latin being a close second [13]. However, it isn't a perfect language and it isn't natural either.

i. *Sanskrit's Efficiency:*

Sanskrit makes use of declensions in nearly every part of speech [14]. This means the ends of words, every single one of them, change depending on the part of speech they're supposed to be. Even proper nouns aren't exempt. The ends of people's names change in a sentence depending on whether they're the subject or the object. This is a bit of a problem for people whose names don't end in a vowel as there is no provision for that in Sanskrit.

The rules of inflections are precise. Just by knowing the ends of a word, one could know its role in a sentence. This makes

word order a non issue. A three word sentence could be written six different ways and a four word sentence in twenty four. None of the permutations would alter their meaning.

Because of the use of declensions, a lot of information is packed in fewer words. This makes transmission of information extremely efficient in speech. Sanskrit is not the only language that can do this though. Latin, an equally dead language, also allowed word order independent sentences in a similar way. Latin too had quite a complicated set of grammar rules. It, like Sanskrit, isn't spoken very much because humans naturally tend to deviate toward simplicity.

ii. *Storage Capacity:*

Despite the arguably best verbal efficiency, there are a few issues with the language in actual knowledge representation. Sanskrit has a glyph based script rather than the alphabet based script as with Latin and its derivatives.

Latin alphabets take one byte of space each. Sanskrit written in the current character set for the Devanagari script is however not an efficient way of storing information. Here's a list of why that is.

- Vowels or consonant glyphs with the inherent vowel takes up 2 bytes of space each.
- The combination of a consonant glyph and a different vowel takes 4 bytes.
- A consonant with a suppressed vowel is 4 bytes.
- A double consonant glyph is 6 bytes.
- A double consonant glyph combined with a vowel is 8 bytes.

Latin script, on the other hand, is consistent. You spend exactly the same number of bytes in conveying a message as the number of letters it contains. My name, 'Denver', takes up 6 characters, 6 keystrokes and 6 bytes in the Latin script. The crude and borderline terrible Devanagari transliteration, 'डेन्वर', takes 3 glyphs (in some renditions, it might look like four – in that case, the two in the middle are a single glyph), 7 careful keystrokes and 14 bytes.

If the character set were redone to start with Devanagari characters rather than the Latin ones, they could reduce space consumption to about a half. Unfortunately, that would mean I'll be spending a byte more to write my name in the wrong script and still have it screw up the pronunciation. Sanskrit is phonemically precise in that the pronunciation of words don't deviate. It does not have a universal phonology. A native speaker of a Sanskrit derived language will find it hard to sound in other languages.

iii. *Sanskrit's Naturalness:*

The fact is that Sanskrit, unlike other languages, hasn't had a natural evolution. Nearly everything about Sanskrit, as is known today, was codified sometime around the year 500 BCE by one person, *Panini*, who was bent on making it as precise and concise as was humanly possible

[15]. Sanskrit didn't simply happen to have the required characteristics of an artificial language by coincidence. It's there by design. It is indeed the work of a primitive computer scientist without the hardware. This is not to say *Panini* intended for his language to be used with machines. At best, his work caught the eye of a pattern seeking human in need of an answer to a difficult, perhaps unsolvable problem – it was bound to happen sooner or later.

The Sanskrit of today, the one reportedly spoken by a few tens of thousands, is about the same as that codified two and a half millennia ago[4]. The language doesn't evolve, it can't evolve. Unlike natural languages, speakers of Sanskrit cannot be classified as proficient or eloquent as its precision does not allow gradations. You either speak the language or you don't; there is no grey. Even artificial languages do not suffer that restriction.

Sanskrit was never widely spoken. During the past two and a half millennia, Sanskrit scholarship was an exclusive club. None other than the Brahmins were allowed to use it. That all literary works in Sanskrit was made accessible only to the Brahmins, spelt its doom [16]. The thing about languages is that, like living organisms, languages too evolve by natural selection.

Natural languages thrive by fitting the need of the era. The flexible of the lot flourish organically forcing the less prominent ones to wither away. Sanskrit's resistance to change was the reason of its demise. This is essentially why every attempt to revive the language will fail, no exceptions.

IV. THE INSIGHT FOCUS ON THE CLAIM:

The paper does not at all contain any claim, mention or indication that Sanskrit can be used as a programming language. In fact, the one and only instance of the word 'program' was in an example sentence meant to illustrate semantic nets. Every single use of the word 'code' or variations thereof have been used to describe sentence construction rules or grammatical syntax. **Notice** that in my systematic summarization of Briggs' piece, the word 'program' doesn't appear once.

The question Briggs tried to answer was whether it was possible for one to create a perfect language for knowledge representation. If a computer scientist were to codify a new language humans could use just as well as a machine, what would the end result look like? He then shows how Sanskrit manages to fulfil all of those requirements. To him, it was astonishing to find that someone who lived a long while ago could accomplish such a feat of brilliance; the entire piece is a recurring acknowledgement of that fact.

Even after all of that, he never once suggested that Sanskrit should be used for knowledge representation. He insisted however that if anyone attempted to create such a language, they would do well to follow a similar pattern of processes as *Panini* did with Sanskrit.

V. SANSKRIT AS A COMPUTER LANGUAGE/NLP:

There is a lot of difference between Sanskrit being suitable for NLP and it being the best or perfect language for computer programming. For Sanskrit to be truly a best language for computer programming, a majority of computer users should be fluent enough in it. Otherwise it is much easier for them to learn a computer friendly language than to learn Sanskrit and then use it for programming a computer. By that logic, we off course fail to see how it is the best language for programming a computer when the number of people who are fluent in C outnumber the number of people who are fluent in Sanskrit. Its natural things but that did not stop some Indians from expanding their chests with pride based on a false premise. They are so lacking in knowledge of history of India, that even when there are other real reasons to proud of Sanskrit such as its influence on modern linguistics or its similarity with Backus-Naur form [17], they ignore them and spout second hand nonsense to inflate,,their,egos.

Perhaps the people who originate these kinds of falsehoods expect everyone all over the world to recognize the technical merits of Sanskrit and throw away their native tongues and instead adopt Sanskrit. If that really is the case, their hypocrisy is mind boggling. These are same people who are likely to make a stink about "western" elements destroying traditional Indian elements. Yet they would like other cultures to throw away their cultural artefacts just because a researcher published a paper saying that Sanskrit is a good language to use in NLP.

It's computer friendly in the sense of Artificial Intelligence not writing a program. Sanskrit is a very predictable language. It is easy to formulate sentences and get meanings from words. It is easy to make words plural. This means that a computer can inherently formulate sentences very easily. Just compare this to English for instance. If we want a computer to formulate sentences we have to teach it all the grammar rules plus the exceptions. For instance, telling the computer that adding "s" to a word does not always mean plural. We have to put in the case of child and children etc. Add to that all the context based meanings and usage of words and it becomes very complicated to program that in. Sanskrit is very formulaic in this sense and the sentences which a computer will formulate given a few set of rules are very accurate.

By the way programming has nothing to do with script. Latin script is mainly used because the West happened to be where computers got popular first so it dominated as choice. All that happens when you load a program is that a bunch of capacitors are charged up. This charging up can be done by any means necessary. With respect to current computer language, if someone ask a question: Is c# better or Java or Python?. But truth is that, every computer language has its advantages. We can't really pick one for all purposes. In this thread, many are either computer illiterates or

programmers commenting on natural language processing. NLP requires both an in-depth knowledge of the language and relevant computer literacy.

Sanskrit is a language that has not been explored for NLP like English or Spanish or even Hindi. Absence of evidence is seldom the evidence of absence. In my opinion, a natural language is more of a computer-buddy if it is simple like you say. The thing with Sanskrit, as I understand is that the all sentences can be very simple, thereby making it easy for the semantic parsers. However, research with proof needs to be presented for which research needs to be done for which some agency should come up with this motives to initiate this research, may required huge amount of grants.

It's not computer friendly or programmer friendly in the practical sense. That's just what people who don't understand this say. The structure of the language is such that it can be easily interpreted by a computer if someone had to make an interpreter for Sanskrit. Knowing Sanskrit won't make you a better programmer.

My opinion, this is one of them shitty myths. What this probably originates from is that, Sanskrit is highly compositional, i.e. there are a few basic concepts and roots which can be combined in many complex ways to come up with new words and ideas. And while there are many "exceptions" to rules and all, it is still very consistent. Think how every rule in English grammar has 100s of common exception, Sanskrit usually doesn't. This is probably because Sanskrit has pretty much always been a "scholarly" language. So, if you want to design some kind of natural language (NLP) system for computers which relies on hard coded rules and all then Sanskrit would probably be a better choice than English or Hindi. But the thing is, for the last three decades, AI researchers have found that the older rule-based paradigms-the Chomsky school of thought has diminishing returns with increasing complexity and moreover is not how human brains work either [18]. So there has been a huge move to statistical methods that rely on huge amount of texts being available. Until 2005 or so, most statistical research also focused on automatically learning "rules" from huge text corpuses, but in the last 10 to 15 years, with the advent of methods like deep learning, we are moving to purely statistical models.

In these modern methods, the size of available text corpus is the most important factor, so Sanskrit doesn't really have any particular advantages. It has very strict grammar rules, which means there are very few or practically no exceptions to the rules. Hence it is easier for artificial intelligence systems or language interpreting compilers to deal with such languages rather than English which has lot more exceptions hence the compiler need to be more complex to understand the language without errors. Actually any Indian language will fare better than English.

But the above restriction is no longer a concern as the computing power is now greatly improved and compiler complexity is not a big issue. Moreover there is no point in using a dead language for computing as you will still have to translate it to a modern language. English is the best choice as it is spoken by many. Infect any Indian language would be a better choice than Sanskrit as the number of speakers will be more.

Those who boast about Sanskrit being the best language for computing should instead implement an artificial intelligence system based on Sanskrit and demonstrate the advantage of Sanskrit vs English with proper benchmarking measurements.

Interpreting the language is only a small component of realizing AI systems. There are many more aspects that are far more complex like understanding a joke, insult, sarcasm etc. And no language will have an edge for these aspects. Hence you don't reduce the AI complexity much by switching to a particular language. Suppose one translator program that can understand Sanskrit and another one that understands English & both have to read the Sanskrit and English versions on the same novel and translate it to a common third language. Measurement of the time taken will settle the debate forever. Unless this is done, such prod claims have the same credibility as the ancient plastic surgeons and head transplants.

When people say computer friendly, they mean it is simpler to create AI generated sentences, without worrying about context or grammar mistake. So, an AI can generate full novel in Sanskrit and it will be free of any grammatical error. Compared to English, there will be less contextual errors as well. So, if a human can understand Sanskrit, computers will find it easier to generate responses using simple rules. Sadly, as for programming, Sanskrit or any natural language will have same problems. So, unless the whole use of computers is to generate a lot of text, any natural language is a disadvantage.

In simpler terms, a novel writing bot will work better in Sanskrit than any contextually complicated language. But that's all the advantage of using Sanskrit. At the same time, disadvantage includes asking users to learn an outdated language, along with reprogramming literally every piece of software we have.

We know that the Sanskrit is the most structured language which has well defined set of rules for grammar & It is also a language which is naturalized by default. These things make it a good language for computer programming. BUT Sanskrit is also one of the most ambiguous languages e.g. What we call tree in English has many words for it in Sanskrit e.g. vriksha, taru, padap, naga, kuja, phalada etc. which makes it not good for programming. So unless this ambiguity is removed, Sanskrit can not be used for computer programming.

If we want to make a computer which could be programmed using vernacular language then in order to

compile that information you'd need to design a whole lot of rules for the computer to interpret everyday sentences and words. The problem with most modern languages is that there aren't many consistencies in the syntax of sentence creation. Therefore its almost impossible to create a perfect compiler to compute vernacular language. However, since Sanskrit and other older languages had a more strictly defined syntax, they are technically more computable. That's not exclusive to Sanskrit though.

We also know that the Computing has two aspects: Logical and Language.

- Logic Language: Logic gives power of motion which leads to the automation of the process just like invention of wheel.
- Language : Language gives the power of expression which provides space for representing the content.

We find contributions of Pāṇini in both the aspects by writing a logical processing machine for generation of Sanskrit words/syntax using a formal language and programming techniques by providing basis to analyse the language and to decode the content coded in it. Natural Languages are the languages that are spoken or written for human communication. Its strength is good for expression of feelings and emotions etc. and its weakness is lead to ambiguity at all levels starting from word, syntax and pragmatics. In the same way the formal Languages are used for representing facts in Science, whose strength is unambiguous syntax and weakness is insubstantial for human beings.

Computing requires unambiguous representation of content. Therefore, the natural choice is formal language. Since Natural Languages lack unambiguous representation, and do not fit into the logic, Artificial Languages are created using formal structure/syntax.

VI. CONCLUSION:

As I have mentioned that, I read the research paper published and I don't remember it mentioning at any single place that Sanskrit could be used as a general-purpose programming language. The paper says knowledge representation in Sanskrit is highly suitable for current on-going work in AI. I have not studied AI or NLP in details, but after reading the paper and couple of Wikipedia links, I also felt that Sanskrit is a good language for AI. In brief, NLP is about human-computer interactions using a common language. We all know that Sanskrit is a natural language. Now, what makes it special that NLP could be achieved with it? The answer is that, any word or sentence in Sanskrit is based on some rules which do not have any exceptions or probably far less exceptions as compared to modern day languages. And no words are added to the language as time progresses unlike other languages. The very first paragraph of the paper seems like it respects all modern languages but don't feel disheartened or frustrated by it. Go on and you will

understand why Sanskrit is associated with NLP. Panini's grammar is what modern linguists call a generative grammar, and that is what they use to describe language grammars themselves; and, as it happens that is also the way computer scientists describe the syntax of programming languages. The ancient Indian grammarian Pāṇini described the structure of Sanskrit using rewriting rules, similar in spirit to the Backus Naur form[19]. With that he established formal language theory, which is used today in research on parsing programming languages. My point of view, it is possible to design a computer entirely based on Sanskrit, but the reasons that the people do not take initiative is due to:

1. No guarantee of success.
2. No one will take up such a project that does not have commercial value. Why would you waste your time in such a project instead of finishing you Phd and getting a job.
3. Huge amount of work involved in this project.

And following points should be kept in mind to initiate the project.

1. Design a Compiler for a programming language based on Sanskrit.
2. Once you have compiler in place you can write minimum Operating System.
3. Define Micro-operations for Computer Hardware System that is based on Sanskrit.

To use Sanskrit would be no different than using English, except in Devanagari script, which would require a whole new series of computers and programming languages to be used. Computers using Sanskrit and Devanagari commands would maybe be more efficient or less efficient. It's a gamble, and a costly and time consuming one at that.

So far, I have found no credible citation stating so. The only things I'm finding are blogs saying Sanskrit is best because Sanskrit sentences have the same meaning whichever order they're read in. I wonder if the people writing these blogs even know how programming languages work. Also, if scientists had to use a heavily inflected language, then they would shift to a completely new script. They could use Latin instead. Almost as inflected as Sanskrit, plus programmers don't have to learn a whole new script, or even discard all current computers and shift to a completely new alphabet system.

Now let's speak in terms of Sanskrit becoming the most suitable NLP language, it would be if only majority of people communicate in Sanskrit. With my experience of AI and Sanskrit, I would say it may be possible. Accordingly to following feasibility point of view:

Pros :

1. Sanskrit is unambiguous in terms of processing. Thanks to the strict grammar rules. So programming an NLP would be much more easier and efficient.

Cons :

1. Sanskrit is not used in day to day basis and almost no one communicates in Sanskrit these days. So, the options of AI for learning is extremely low. Only if 25% of people in the world learn Sanskrit and communicate to AI, it would learn only a portion of whole world's activity through its rules.
2. We need processors to process Natural Language, to do that, we need people who communicates in Sanskrit and has a good hold over it.
3. Sanskrit NLP will be only a matter of research until and unless everyone communicates in Sanskrit.

So, you can see that, using a NLP Translator for Sanskrit, you can only make it useful if the whole world or at least 25% of its population, communicate in Sanskrit.

But in this era of internet, people already have developed a big portion of NLP based on English, so using Sanskrit as a base for NLP is still a question.

Till a Properly Created Sanskrit NLP is made, it will be difficult to assess whether it is the Best Language, Sanskrit is a Precursor Language and will probable not evolve in the same way that Data and AI evolve.

The article published by a Mr. Briggs has been typed again and again and certain sites have even propagated it saying that NASA uses Sanskrit as a Language for their computers. There is no proof for the same, other than the same article. It therefore should be awaited, that If and When someone develops a Sanskrit NLP in a usable state, One would have to see the benefits of the same. The paper doesn't conclude as wether the Sanskrit is not good or bad for AI. Instead it says, even AI can be solved if rethought in line with the structure of Sanskrit.

One of the conclusions I draw personally is that IF the knowledge (currently in latin, french, german, italian or other languages) were made available in a strict rules-based language (like Sanskrit) it may make it easier/optimal for an AI system to process, decipher & utilize the knowledge to take decisions.

ACKNOWLEDGEMENT

REFERENCES

[1]. https://www.aaai.org/ojs/index.php/aimagazine/article/viewArticle/466
[2]. https://www.computerhope.com/jargon/p/proglang.htm
[3]. https://www.nasa.gov/
[4]. https://www.thefreedictionary.com/artificial+intelligence
[5]. https://scepthink.com/2015/01/18/sanskrit-computer-programming
[6]. http://calmerthanyouare.org/2015/03/19/betteridges-law.html
[7]. https://psychologydictionary.org/artificial-language/
[8]. http://intelligence.worldofcomputing.net/knowledge-representation/semantic-nets.html#.WhPMU0qWbIU
[9]. http://nptel.ac.in/courses/106105078/pdf/Lesson%2019.pdf
[10]. http://users.cs.cf.ac.uk/dave/AI2/node60.html
[11]. https://www.quora.com/Which-language-has-the-highest-consonant-to-vowel-ratio
[12]. https://www.techopedia.com/definition/31856/semantic-element
[13]. http://www.academicroom.com/topics/what-is-sanskrit
[14]. https://scepthink.com/2015/01/18/sanskrit-computer-programming/
[15]. http://www-history.mcs.st-andrews.ac.uk/References/Panini.html
[16]. https://aamjanata.com/religion-philosophy/sanskriti-sanskrit-language-brahmins/
[17]. https://www.quora.com/What-is-the-link-between-the-Backus-Naur-form-and-Sanskrit-and-its-relevance-to-computer-programming-languages
[18]. https://www.sott.net/article/328813-Evidence-rebuts-Chomskys-theory-of-language-learning
[19]. http://matt.might.net/articles/grammars-bnf-ebnf

Books:

[20]. G Cardona, *Panini : a survey of research*, Paris,1976.
[21]. G G Joseph, *The crest of the peacock* ,London,1991.
[22]. K. Chandrasekharan,Brahmasri,V.H.Subrahmanya ,Sanskrit Literature,The International book house,Ltd, Bombay,1951.
[23]. William Dwight Whitney, Sanskrit Grammar Dover Publications,2003.
[24]. Walter Harding Maurer, The Sanskrit Language: An Introductory Grammar And Reader, Routledge,1995 .

Articles:

[25]. P Z Ingerman, 'Panini-Backus form', *ACM* **10** (3), 1967.
[26]. Chandana Bathulapalli, Drumil Desai andManasi Kanhere, Use of Sanskrit for natural language processing, **IJSR 2016; 2(6): 78-81.**
[27]. V. Zue and J. Glass, "Conversational Interfaces: Advances and
[28]. Challenges" Proceedings of the IEEE, Special Issue on Spoken Language Processing, Vol. 88, August 2000.
[29]. J. Glass and S. Seneff, "Flexible and Personalizable Mixed-Initiative Dialogue Systems," presented at HLT-NAACL 2003 Workshop on Research Directions in Dialogue Processing, Edmonton, Canada, ay2003.
[30]. V. Zue, et al., "JUPITER: A Telephone-Based Conversational Interface for Weather Information," IEEE Transactions on Speech and Audio Processing, Vol. 8 , No. 1, January 2000.