Dynamic Construction of Telugu Speech Corpus for Voice Enabled Text Editor

Dr. K. V. N. Sunitha *k.v.n.sunitha@gmail.com*
Principal,
BVRIT Hyderabad College of Engg. for women,
Bachupally, Hyderabad, A.P., India

A. Sharada *sharada.nirmal@gmail.com*
Assoc.Prof, CSE Dept
G.Narayanamma Inst.Of Technology & Science
Shaikpet, Hyderabad,A.P., India

Abstract

In recent decades speech interactive systems have gained increasing importance. Performance of an ASR system mainly depends on the availability of large corpus of speech. The conventional method of building a large vocabulary speech recognizer for any language uses a top-down approach to speech. This approach requires large speech corpus with sentence or phoneme level transcription of the speech utterances. The transcriptions must also include different speech order so that the recognizer can build models for all the sounds present. But, for Telugu language, because of its complex nature, a very large, well annotated speech database is very difficult to build. It is very difficult, if not impossible, to cover all the words of any Indian language, where each word may have thousands and millions of word forms. A significant part of grammar that is handled by syntax in English (and other similar languages) is handled within morphology in Telugu. Phrases including several words (that is, tokens) in English would be mapped on to a single word in Telugu.Telugu language is phonetic in nature in addition to rich in morphology. That is why the speech technology developed for English cannot be applied to Telugu language. This paper highlights the work carried out in an attempt to build a voice enabled text editor with capability of automatic term suggestion. Main claim of the paper is the recognition enhancement process developed by us for suitability of highly inflecting, rich morphological languages. This method results in increased speech recognition accuracy with very much reduction in corpus size. It also adapts Telugu words to the database dynamically, resulting in growth of the corpus.

Keywords: Speech Corpus, Suggestion List, Text Editor, Signal Comparator, Incremental Growth.

1. INTRODUCTION

A corpus is a large and representative collection of language material stored in a computer processable form. Corpora provide realistic, interesting and insightful examples of the language use for theory building and for verifying hypothesis. Insights obtained from analysis of corpora have led to fresh and better understanding of how language actually works. A lot of research is carried out throughout India over the decade and most of the documents are being prepared in local language through software available for the purpose. Though there are many software's and keyboards available to produce such documents, the accuracy is not always acceptable, the chance of getting errors is more.' particularly for Indian language most of which are highly inflecting. Developing a robust voice enabled editor to transcribe continuous speech signal into a sequence of words is a difficult task, as continuous speech does not have any natural pauses in between words. It is also difficult to make the system robust for speaker variability and the environment conditions.

The conventional method of building a large vocabulary speech recognizer for any language uses a top-down approach to speech. Top-down approach means system first hypothesize the sentence, then the words that make up the sentence and ultimately the sub-word units that make up the words. This approach requires large speech corpus with sentence or phoneme level transcription of the speech utterances. The transcriptions must also include different speech order so that the recognizer can build models for all the sounds present. It also requires maintaining a dictionary with the phoneme/sub word unit transcription of the words and language models to perform large vocabulary continuous speech recognition. The recognizer outputs words that exist in the dictionary. If the system is to be developed for a new language it requires building of a dictionary and extensive language models for the new language. In country like India which includes 22 officials and a number of unofficial languages, building huge text and speech databases is a difficult task.

The paper is organized as follows: Section 2 explains the state of the art, Section 3 details the characteristics of Telugu Language, Section 4 describes the proposed model in which ASR is the major contribution, Section 5 elaborates on Architecture of ASR that is designed to suit Indian languages and enhances speech recognition accuracy, Section 6 explains Speech correction procedure that enhances speech recognition accuracy, Section 7 gives Results and Conclusion.

2. LITERATURE SURVEY

There has been a significant progress in advancing Automatic Speech Recognition (ASR) technologies in the past several decades. However, ASR systems are not widely adopted today. Among all the issues that prevent users from using ASR systems, recognition accuracy is still the most important factor [6]. It is well known that an ASR system adapted to a specific user performs better. It's not sufficient by its own due to the fact that there are more elements to be adapted than the AM alone. For example, the default dictionary does not include some of the words frequently used by a specific user. Examples of these Out Of Vocabulary (OOV) words are project names, acronyms, and foreign names. Adapting the lexicon would recover on average 1.2 times as many errors as OOV words removed [7].

The understanding of importance of access to a large amount of correctly annotated speech data is not only widely recognized among the people working in the field of speech recognition, but became generally recognized in the whole speech researchers' society. For this reason nowadays the increasing number of professionals engaged in speech research is involved in the projects, which aim the creation of large-scale speech databases [31]. While corpus based and statistical approaches to speech have been well established elsewhere in the world , India is still lagging far behind. A more recent development was a Hindi recognition system from HP Labs, India which involved Hindi speech corpus collection and subsequent system building [23] . But for Telugu language, the work is started recently and a large, well annotated speech database is almost not available. The reason being the complexity of Telugu language and the difficulties in developing the speech database. In India, CDAC, LDC-IL(CIIL), IISc, IBM, TIFR, IIT –Mumbai including few other prestigious organizations are working on speech Technologies and Speech corpus creation for Indian Languages.

The approach proposed here is different from the research work mentioned in [17], [19], [21], in that they work at the signal level, where as our approach exploits the language characteristics and works at Language Model.

3. CHARACTERISTICS OF TELUGU LANGUAGE

Telugu is a Dravidian language spoken in southern Indian states and is the official language of Andhra Pradesh. It is considered to be the 2nd most widely spoken language in India after Hindi. There are 75 million first language and 5 million 2nd language speakers of Telugu [6]. The most important feature of Telugu language is its phonetic nature. There is almost one to one correspondence between what is written and what is spoken in contrast to English.

For xample, in English we write l-a-u-g-h but pronounce it as l-a-f. That is, we are writing 'g' sound but saying 'f' sound. This kind of situation does not occur in Telugu language. What we write is what we speak. The rules required to map the letters to sounds in Telugu are almost straight forward[8].

A significant part of grammar that is handled by syntax in English (and other similar languages) is handled within morphology in Telugu (and other Dravidian languages). Phrases including several words (that is, tokens) in English would be mapped on to a single word in Telugu. Verbs may include aspectual auxiliaries apart from tense and agreement. There are several types of non-finite forms too. A single verbal root can lead to formation of a few hundred thousand word forms. Nouns are also inflected for number and case. Derivation being very productive, even more forms become possible when we consider full word forms. These words are made up of several morphemes conjoined through complex morpho-phonemic processes.

In inflectional language every word consists of one or several morphemes into which the word can be segmented; consider for instance the morpheme segmentations of the following Telugu words: “Ame(she), Ame+yokka(of her), Ame+tO(with her), AmE+nA(is it she)”. In highly-inflecting and compounding languages the number of possible word forms is very high. This poses special challenges to NLP systems dealing with these languages. For example, in automatic speech recognition it is customary to use pre-made lists of attested word forms as a “normative” vocabulary.

The incoming acoustic signal is matched against the list, and only words contained in the corpus can be recognized. Such a word list can be created by collecting word forms from large text corpora or existing lexicons, and the aim is to obtain as much coverage as possible of the words of the language. When processing languages with extremely rich word forming like Telugu, the resulting word lists are typically very large, this is demanding, from a computational point of view. A more serious problem is that many perfectly valid word forms are likely to be missing from the list anyway, since they might never have occurred in the corpus used as a source. For words which are out of the dictionary the recognition accuracy is quite low and gets matched to the nearest possible word in the dictionary.

4. PROPOSED MODEL

One dimension of variation in speech recognition tasks is the vocabulary size. Speech recognition is easier if the number of distinct words we need to recognize is smaller. So tasks with a two word vocabulary, like *yes* versus *no* detection, or an eleven word vocabulary, like recognizing sequences of digits, in what is called the digits task, are relatively easy. On the other end, tasks with large vocabularies, like transcribing human-human telephone conversations, or transcribing broadcast news, tasks with vocabularies of 64,000 words or more, are much harder.

We propose a method where we use an efficient ASR technique to recognize the word. The given word is compared with the words in the database. If an exact match is found that is displayed. Otherwise, Speech correction procedure designed by us will be applied and a list of suggested words is given. If the user accepts any of those suggested words then it is displayed in the editor. Otherwise the word is added to the database by taking 3 samples from the user.

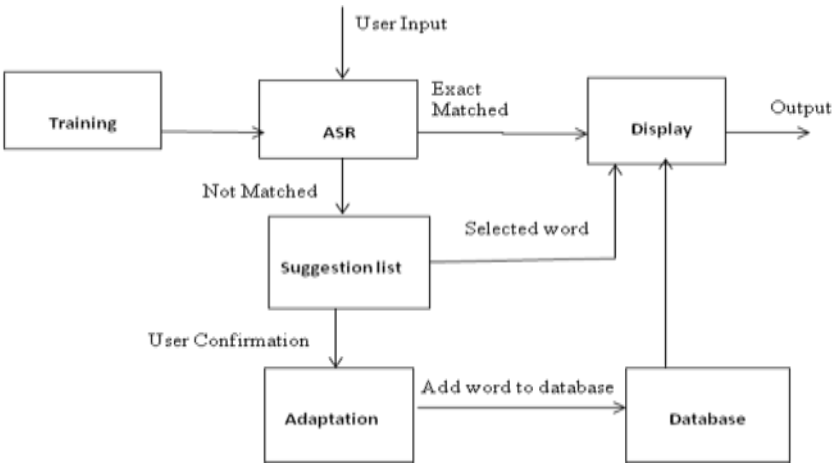

FIGURE 1: Proposed Model

Steps for Corpus Construction

The proposed approach is a three-step process:

- recognition of uttered word
- generating list of probable words in case of the word does not match with the existing words of database
- updating the database with newly added words. The system will check whether the error is likely caused by OOV words. If yes, the new words are added into the lexicon.

4.1 Training

In this module, the system is designed to take three speech samples of a word from the user and thus the system is trained so that it recognizes the word when uttered for the next time. This phase involves extracting MFCC features of these sample speech and an average model for these samples is built. Thus the system is trained for each new word that to be added.

Steps:

- Collecting speech samples
- Feature Extraction
- Finding templates of each sample
- Finding templates across all the samples

The training phase is implemented by using two algorithms. They are as follows:

- Segmental K-Means algorithm
- Baum-Welch Algorithm

4.2 Automatic Speech Recognition

This module is the major claim of our paper. We designed an efficient ASr that suits highly inflecting languages. Architecture of our ASR is explained in detail in Section 5.

4.3 Suggestion List

In case, the user spoken word is not recognized, then Speech correction Procedure developed by us will be applied. Based on the probabilities, the suggested word list is generated, from which user can select a word. This can be done in two ways. The first approach is, distance

computation of signals of corpus words with the user input word. And the second approach is word co-occurrence probabilities are computed and stored beforehand. Later approach requires deep insight into the language under consideration whereas first approach is independent of language.

4.4 Adaptation

If the user input is not matched with any of the words present in database, then a list of possible words are suggested that closely matches with the user input. If the user is does not accept any words from the suggested list, then the uttered word should be added to the database. For this user is asked to repeat the utter the same word for three times. Features are extracted and model is built for that word. Then it is added to the database.

4.5 Database

As the proposed system is aimed at dynamic recognition and insertion of speech input, we use Trie data structure for efficient and faster searching. Trie is a data structure that can be used to do a fast search in a large text and that stores the information about the contents of each node in the path from the root to the node, rather than the node itself. A Trie node, named for its successful use in information retrieval, is an array of pointers, one for each character in an alphabet. Each leaf node is the terminus of a chain of trie nodes representing a string. For string management, tries are fast with reasonable worst-case performance. A *Trie* (short for re*Trie*val) is a multiway tree that is used for efficient searching. The main objective of this work is to **reduce the search space and minimize the data base** and thereby improve the performance using Trie based data structure.

Our database structure is explained in detail in section 5.1.

5. ARCITECTURE OF *ASR* IN THE PROPOSED MODEL

Factored language models have recently been proposed for incorporating morphological knowledge in the modeling of inflecting language. As suffix and compound words are the cause of the growth of the vocabulary in many languages, a logical idea is to split the words into shorter units.The approach used here aims at reducing the above mentioned problem of having a very huge corpus for good recognition accuracy. It exploits the characteristic of Telugu language that every word consists of one or several morphemes into which the word can be segmented. The base ASR identifies the nearest root word, over which Speech correction procedure is applied.The proposed model consists of the Stem Recognizer, Segmenter, Database and Signal comparator modules.

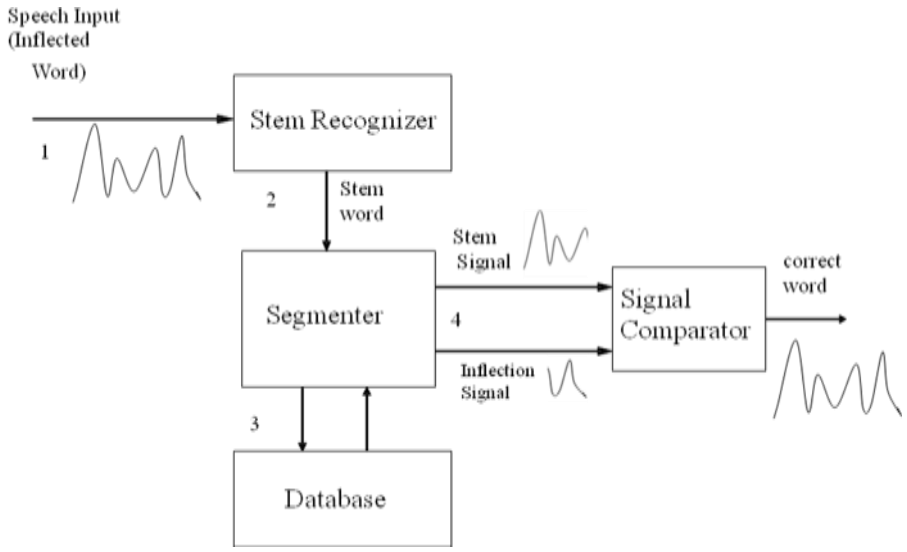

FIGURE 2: ASR Architecture

5.1 Database

Database contains stems and inflections separately. It does not store inflected words as it is very difficult, if not impossible, to cover all inflected words of the language. The database consists of 2 dictionaries:

a) Stem Dictionary

b) Inflection Dictionary

Stem dictionary contains the stem words of the language, signal information for that stem which includes the duration and location of that utterance and list of indices of inflection dictionary which are possible with that stem word.

Inflection Dictionary contains the inflections of the language, signal information for that inflection which includes the duration and location of that utterance.

Both the dictionaries are implemented using trie structure in order to reduce the search space. Details of this implementation can be seen in [3].

Word	Stem signal information	Inflection Indices
amma (అమ్మ)	D:\work\s1.wav	1, 2, 4, 6,7,8
anubhUti(అనుభూతి)	D:\work\s2.wav	1,7
ataDu (అతడు)	….	….

TABLE 1: Stem Dictionary

Sl.No	Inflection	Inflection Signal Information
1	ki (కి)	D:\work\inf1.wav
2	tO (తో)	D:\work\inf2.wav
3	guriMci (గురించి)	
4	….	....

TABLE 2: Inflection Dictionary

As the vocabulary of Telugu language is infinite, it aims at collecting the most frequently used words and stores to database in the form of audio files.

5.2 Stem Recognizer

This module identifies the nearest stem of the given speech input. This is a basic ASR system whose accuracy may not be very good. Subsequent modules will help enhance this accuracy.

5.3 Segmenter

Segmenter takes the identified stem word as input. Accesses the database for signal information and segments the given signal into stem and inflection. If the utterance is a stem word then the second part will be empty.

5.4 Signal Comparator

This module compares the inflection part of the signal with the possible inflections list from the database and gives correct inflection. This will be given to Morph Analyzer to apply morpho syntactic rules of the language and gives the correct inflected word.

6. SPEECH CORRECTION PROCEDURE

Speech correction procedure is explained in Fig 3 shown below.

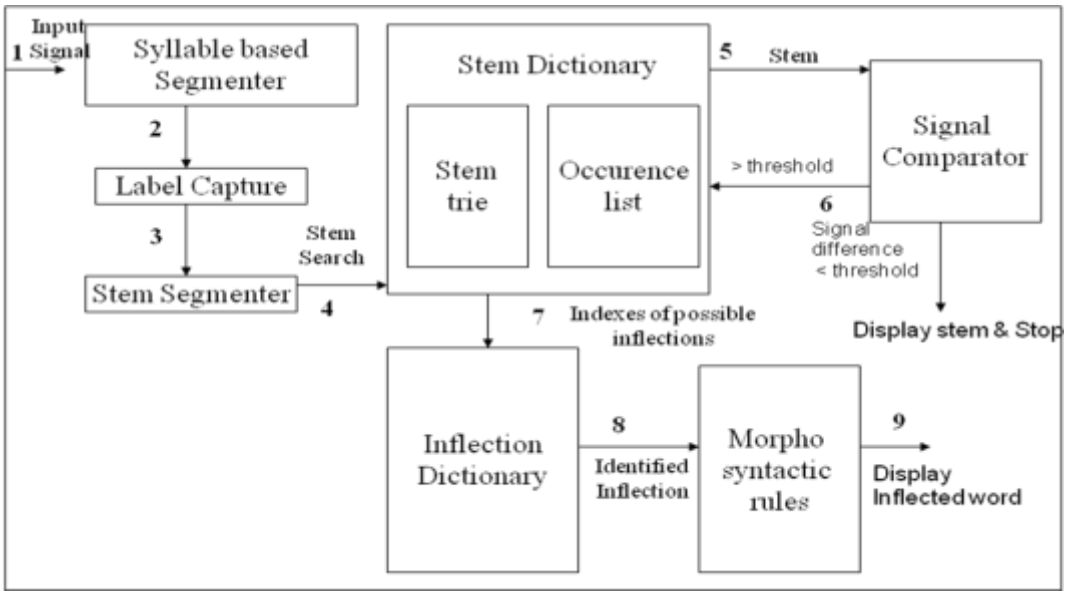

FIGURE 3: Speech Correction Procedure

The steps of the procedure are listed as follows:

***Step 1*:** Capture the utterance

***Step 2*:** Get the nearest stem word

***Step 3*:** Segment the signal into stem and inflection

***Step 4*:** Get its syllabified form

***Step 5*:** Get the inflection information

***Step 6*:** Compare the inflection signals possible with that stem one by one and store the signal differences into an array

***Step 7*:** Apply morpho syntactic rules of the language to combine stem and inflection

***Step 8*:** Display the inflected word

Using the rules the possible set of root words are combined with possible set of inflections and the obtained results are compared with the given user input and the nearest possible root word and inflection are displayed if the given input is *correct*.

If the given input is *not correct* then the inflection part of the given input word is compared with the inflections of that particular root word and identifies the nearest possible inflection and combines the root word with those identified inflections, applies sandhi rules and displays the output.

The user input is syllabified and this would be the input to the analyzer module. Matching the syllabified input from starting with the root words stored in dictionary module a possible set of root words is obtained.

When there is more than one root word or more than one inflection has minimum edit distance then the model will display all the possible options. User can choose the correct one from that. E.g., when the given word is ***pustakaMdO*** (పుస్తకందో), the inflections ***tO*** making it ***pustakaMtO*** (పుస్తకంతో) meaning 'with the book' and ***lO*** making it ***pustakaMlO*** (పుస్తకంలో) meaning 'in the book') mis are possible. Present work will list both the words and user is given the option. We are working on improving this by selecting the appropriate word based on the context.spelled

The recognized stem is syllabified and this would be the input to the analyzer module. Matching the syllabified input from starting with the root words stored in dictionary module a possible set of root words is obtained.

For eg**:** ***nAnnagAriki*** ← user input

(Identified stem after syllabification)

⇓

nAn-na

Once possible root words identified the inflection part is compared in the reverse direction for a match in the inflection dictionary. It will consider only the inflections that are mentioned against the possible root words, thus reducing the search space and making the algorithm faster.

gA-ri-ki ⟸ (matching from reverse)

After getting the possible set of root words and possible set of inflections they are combined with the help of SaMdhi formation rules.

Here in this example ***gA-ri-ki*** is compared with the inflections of the root word ***nAnna.*** After comparing it identifies ***gAriki*** as the nearest possible inflection and combines the root word with the inflection and displays the output as ***"nAnnagAriki".***

7. SAMPLE RESULTS & CONCLUSION

The approach proposed here results in increased speech recognition accuracy with very much reduction in corpus size. This approach is different from the research work mentioned in [17], [19], [21], in that they work at the signal level, where as our approach exploits the language characteristics and works at Language Model. It also adapts Telugu words to the database dynamically, resulting in growth of the corpus.

It eliminates the need for a huge corpus, the main hurdle of ASR technology development for Indian languages. *This method recognizes all inflected word forms just by using k+j words in the database which otherwise requires k*j, where 'k' is number of stem words and 'j' is number of inflections possible in the language.* The model proposed is very useful for enhancing speech recognition accuracy.

When there is more than one root word or more than one inflection has minimum edit distance then the model will display all the possible options. User can choose the correct one from that. We are working on improving this by selecting the appropriate word based on the context. This approach not only identifies all inflected word forms with a small database but also corrects the wrongly uttered/recognized word to its correct-form. Sample result can be seen in Table 3.

This work can contribute to the future employment of speech technologies in a variety of possible applications like keyword based search for given acoustic signal which at present is done by giving text as input, and generating documents through speech rather than through keyboard. We hope this work helps in accelerating research work in Telugu Speech Technoloy, by eliminating the need of laborious task of constructing huge speech corpus.

Input word	Nearest Root word	Nearest Inflection	Is Correct	Suggestion List	Select/Add
pustakAlu	pustakaM	lu	yes	---	---
pustakAdu	pustakaM	du	no	pustakAlu	Select
pustakaMdO	pustakaM	lO tO	no	pustakaMlO pustakaMtO	Select
putakaM	patakaM (పతకం) pustakaM	---	no	patakaM pustakaM	Select
puste	pustakaM	---		pustakaM	Add 'puste' to database
pusteki	puste	ki	yes	---	---
pusta	puste	---	no	puste	Select

Table 3: Sample Result.

8. REFERENCES

1. "LANGUAGE IN INDIA Strength for Today and Bright Hope for Tomorrow", Vol 6, August 2006.

2. "Dravidian Phonological Systems", University of Washington Press, 1975.

3. Oppenheim, A. V., & Schafer, R.W. (1975). "Digital signal processing", Englewood Cliffs: Prentice-Hall.

4. "Conversational Telugu", N.D.K.Institute of Languages, 1992.

5. "Issues in Indian languages computing in particular reference to search and retrieval in Telugu Language", Emerald Group Publishing Ltd.

6. Jinxi Xu and W. Bruce Croft."Corpus based stemming using co-occurrence of word variants". *ACM Trans.Inf.Syst.,* 16(1):61-81,1998.

7. J.L.Dawson. "Suffix removal for word conflation". In *Bulletin of the Association for Literary and Linguistic Computing*, volume 2(3), pp. 33-46, Michaelmas,1974.

8. K.V.N.Sunitha, A.Sharada, "Telugu Text Corpora Analysis for Creating Speech Database", *IJEIT*,ISSN 0975-5292, Dec 2009, Volume 1, No.2.

9. L. Deng, and X. Huang, "Challenges in Adopting Speech Recognition", *Communications of ACM*, vol. 47, No. 1,pp69-75, Jan 2004.

10. CampbellWN, Isard S D "Segment durations in a syllable frame", *J. Phonetics: Special issue on speechsynthesis* 1991 19: 37–47.

11. Young, S., and G. Bloothooft. eds. 1997.*Corpus-Based Methods in Language and Speech Precessing.*Vol-II.Dordrecht: Kluwer Academic Publishers.
12. D. J. Ravi and Sudarshan Patilkulkarni, "A NovelApproach to Develop Speech Database for Kannada Text-to-Speech System", *Int. J. on Recent Trends in Engineering & Technology*, 2011, Vol. 05, No. 01, in ACEEE.

13. Atkins, S., J. Clear and N. Ostler. 1992. "Corpus Design Criteria." *Literary and Linguistic Computing* . 7(1): 1-16.

14. Barlow, M. 1996. "Corpora for Theory and Practice." *International Journal of Corpus inguistics,* 1(1): 1-38.

15. M.A. Anusuya · S.K. Katti, "Front end analysis of speech recognition: a review", *Int J Speech Technol* (2011) 14: 99–145.

16. K Subramanyam, D.Arun Kumar*,"*Static Dictionary for Pronunciation Modeling", *IJRET*,Oct 2012, Volume: 1 Issue: 2 , pp.185 – 189.

17. N. Usha Rani and P.N. Girija, "Analyzing and Correction of Errors to Improve the Speech Recognition Accuracy for Telugu Language", *CiiT International Journal of Artificial Intelligent Systems and Machine Learning ,* Issue : June 2011.

18. Pukhraj P Shrishrimal, Ratnadeep R Deshmukh and Vishal B Waghmare. "Article: Indian Language Speech Database: A Review", *International Journal of Computer Applications* 47(5):17-21, June 2012. Published by Foundation of Computer Science, New York, USA.

19. S. Lokesh, G. Balakrishnan, "Speech Enhancement using Mel-LPC Cepstrum and Vector Quantization for ASR", *European Journal of Scientific Research* ISSN 1450-216X Vol.73 No.2 (2012), pp. 202-209.

20. Prateek Srivastava, Reena Panda&SankarsanRauta, "A Novel, Robust, Hierarchical, Text-Independent Speaker Recognition Technique", *Signal Processing: An International Journal* (SPIJ), Volume (6) : Issue (4) : 2012.

21. Urmila Shrawankar & Vilas Thakare, "Parameters Optimization for Improving ASR Performance in Adverse Real World Noisy Environmental Conditions", *International Journal of Human Computer Interaction* (IJHCI), Volume (3) : Issue (3) : 2012 58.

22. M.S. Salam, Dzulkifli Mohamad and S.H. Salleh, "Improved Statistical Speech Segmentation Using Connectionist Approach", *Journal of Computer Science* 5 (4): 275-282, 2009 ISSN 1549-3636

23. Gopalakrishna~Anumanchipalli et. al., "Development of indian language speech databases for large vocabulary speech recognition systems", *Proceedings of International Conference on Speech and Computer(SPECOM)*, Patras,Greece, Oct 2005.

24. R.Krovetz. "Viewing morphology as an inference process". In *Proceedings of Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 191-203, 1993.

25. Paice C and Husk G. "Another Stemmer". In *ACM SIGIR Forum* 24(3):566,1990.

26. L. Lamel and G. Adda, "On Designing Pronunciation Lexicons for Large Vocabulary,Continuous Speech Recognition", *Proc. International Conference on SpokenLanguage Processing (ICSLP'96)*, pp6-9, 1996.
27. Khan A N, Gangashetty S V, Yegnanarayana B "Syllabic properties of three Indian languages: Implications for speech recognition and language identification", In *Int. Conf. Natural Language Processing*, Mysore, India, pp. 125–134, 2003.

28. Agrawal S. S., "Recent Developments in Speech Corpora in Indian Languages: Country Report of India", *O-COCOSDA 2010*, Kathmandu, Nepal.

29. Chalapathy Neti, Nitendra Rajput, Ashish Verma, "A Large Vocabulary Continuous Speech Recognition system for Hindi", In *Proceedings of the National conference on Communications*, 2002 , Mumbai, pp. 366-370.

30. M.F.Porter. "An algorithm for suffix stripping".In *readings in information retrieval*, pages 313-316, San Francisco,CA,USA,1997.Morgan Kaufmann Publishers Inc.

31. Akshar Bharathi, Prakash~Rao K, Rajeev Sangal, and S.M.Bendre, "Basic statistical analysis of corpus and cross coparision among corpora", *Technical Report~4*, IIIT, Hyderabad, www.iiit.net/techreports/2002.4.pdf, 2002

32. vishwabharat@tdil, MIT Govt. of India Magazine

33. Size of Speech Corpora (As on Dec 2011) , Available at: *http://www. ldcil. org/resourcesSpeechCorp. Aspx.*

34. K.Nagamma Reddy, "Phonetic, Phonological, morpho-syntactic and semantic functions of segmental duration in spoken Telugu:acoustic evidence".
35. Bansal, R.K. 1969.*The intelligibility of Indian English*. *Monograph No. 4*, CIEFL, Hyderabad.
36. B.Ramakrishna Reddy, "Localist studies in Telugu Syntax", *Ph.D Thesis*, University of Edinburgh, 1976.