Automatic Extraction of Phonetically Rich Sentences from Large Text Corpus of Indian Languages

Karunesh Arora, Sunita Arora, Kapil Verma, S. S. Agrawal
Natural Language Processing Division, Centre for Development of Advanced Computing, Noida
karunesharora@cdacnoida.com

Abstract
A set of phonetically rich sentences is a requirement for representing different speech units, to be used for developing Automatic Speech Recognition and Speech Synthesis Systems. Selecting such a set from a large text corpus without modifying the characteristics of the corpus is still a difficult task. A major concern in this process is to decide on what basis sentences must be chosen so that it covers all phonetic aspects of the language under study in a minimum possible size. This paper describes a simple process of automatically extracting such set of sentences from a large text corpus of a given Indian Language and also presents an algorithm for the process. The process discussed in this paper is language independent and works for most of the Indian Languages. The extent of success, in terms of phonetic richness of the sentences, achieved in the process is also discussed.

1. Introduction
A set of phonetically rich sentences is a set of sentences that closely resemble the phonetic characteristics of the language under study. It is not practical to use a large set of data for automatic speech recognition and synthesis, thus the need to develop such set arises. Some similar work has been done for different European and other languages like Chinese [1] etc. to extract phonetically rich sentences but the amount of work done for Indian Languages is of limited scope. A major difficulty in this process is to decide the criteria on which these sentences should be selected and the characteristics (size, domain, etc) of the text corpus taken as input to the system.

The process of automatically extracting phonetically rich sentences involves, analyzing a large corpus and then extracting the sentences from it using sentence extraction criteria, based on some statistical analysis. As most of the Indian languages scripts are orthographic representation of speech sounds and share a common phonetic base, a large text corpus of Indian languages has been analyzed to extract phonetically rich sentences. A brief description of the corpus is given in next section. The corpus is statistically analyzed using a Statistical Analysis Tool (Vishleshika [2]). Vishleshika performs several types of analysis ranging from simple frequency counts to extraction of linguistic features.

Following is a brief introduction to characteristics of Indian Languages on the basis of which our system is designed.

1.1 Phonetic Nature of Indian Languages
All Indian languages scripts are phonetic in nature and thus share a common phonetic base. The basic unit of Indian languages scripts is characters, which are orthographic representation of speech sounds. Classification of Hindi consonants and vowels are presented below.

<table>
<thead>
<tr>
<th>Sgn &amp; Afferents</th>
<th>Glides</th>
<th>Fricatives</th>
<th>Stop &amp; Affricates</th>
<th>Nasals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voiced</td>
<td>ㄹ, ㅁ, ㅂ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ</td>
<td>ㄹ, ㅁ, ㅂ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ</td>
<td>ㅍ, ㅊ, ㅌ, ㅎ, ㅋ, ㅌ, ㅍ, ㅎ</td>
<td>ㅌ, ㅍ</td>
</tr>
<tr>
<td>Voiceless</td>
<td>ㄹ, ㅁ, ㅂ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ</td>
<td>ㄹ, ㅁ, ㅂ, ㅈ, ㅊ, ㅋ, ㅌ, ㅍ, ㅎ</td>
<td>ㅍ, ㅊ, ㅌ, ㅎ, ㅋ, ㅌ, ㅍ, ㅎ</td>
<td>ㅌ, ㅍ</td>
</tr>
</tbody>
</table>

Table 1: Classification of Hindi Consonants [3]

<table>
<thead>
<tr>
<th>Front</th>
<th>Middle</th>
<th>Back</th>
</tr>
</thead>
<tbody>
<tr>
<td>/t/</td>
<td>/k/</td>
<td>/d/</td>
</tr>
<tr>
<td>/d/</td>
<td>/g/</td>
<td>/b/</td>
</tr>
</tbody>
</table>

Table 2: Classification of Hindi Vowels

1.2 Unit Selection
To extract phonetically rich sentences we have considered syllables as the basic unit and used syllabification rules to syllabify the text. A syllable is a unit of speech that is made up of one or more phones (single sounds or "phonetic segments") and in turn makes up words. It influences the rhythm of a language, its prosody, its poetic meter, its stress patterns, etc.

The general structure of a syllable consists of three parts: the onset, the nucleus, and the coda. The nucleus is usually a vowel or a diphthong. The onset is what comes before the nucleus, and the coda is what comes after it. Hindi language allows syllables with empty codas (i.e. no consonants after the nucleus), and empty onsets. A syllable of the form CV (consonant + vowel, with an empty coda) is called an open
syllable, while a syllable that has a coda (CVC, etc.) is called a closed or checked syllable.

In Hindi, words are composed of syllables of type \( C_0 V C_0 \) [4] e.g. /kam\(\)a\(\)/ (Lotus) consists of syllables /ka/ and /ma/ of type CV and /strim/ (womanhood) consist syllable /str/ of type CCCV.

For in-depth analysis, we are also taking the position of the syllable into consideration, at which it appears in a word e.g. in /kam\(\)/ (work), /man\(\)/ (house), /ko\(\)ka\(\)/ (his/her) and /ka/ (of) syllable /ka/ appears at starting, middle, end and isolated positions in the word respectively.

2. Statistical Analysis

Statistical analysis is performed on a large set of data from GyanNidhi Corpus, which is parallel in multiple Indian languages. GyanNidhi contains variety of data from books published by National Book Trust India, Sahitya Akademi, Navjivan Publications, Publications Division, Sri Aurobindo Ashram, as they publish books covering various domains, in most of the Indian languages. The extraction of phonetically rich sentences experiment has been carried out on a subset of this corpus that contains approximately 5147 Sentences containing 46421 words in Hindi Language. These sentences are chosen from different domains having a simple construct.

Following is the detailed process flow and the methodology of the system.

2.1. Algorithm

Figure 1. Algorithm for Automatic Extraction of Phonetically Rich Sentences from Large Text Corpus of Indian Languages
2.2. Methodology

For selecting the sentences from above mentioned text corpus, a “Unique Unit Coverage Score (Sc)” is calculated as:

\[ Sc = \frac{Uc}{Ut} \]

where \( Uc \) is the number of unique units covered in the sentence and \( Ut \) is the total units in the sentence.

Details of the actual algorithm are as follows:

i. Given a text corpus, average units per sentence \( (Ua) \) is calculated as follows:

\[ Ua = \frac{Nu}{Ns} \]

where \( Nu \) is the total number of unique units in corpus and \( Ns \) is the total number of sentences in the corpus.

ii. Sequentially all sentences are marked with “Unique Unit Coverage Score (Sc)” and assigned a weight \( Ws \) such that:

\[ Ws = \begin{cases} Sc + 1 & \text{if } Ua - 5\% < Ut < Ua + 5\% , \\ \frac{Sc + 0.5}{Sc} & \text{otherwise} \end{cases} \]

\( Ws \) are used to confine the selected sentences to the desired length (i.e. close to the average length of the sentences in the corpus). In this step, too long or too short sentences have been assigned lesser weight than the sentences that have the length close to the average sentence length \( (Ua) \). This assures that sentences that are close to average length get high priority.

iii. In the sequential selection, a sentence is selected only if it contains at least one unique instance of a unit not present in the previously selected sentences. This approach helped in reducing the size of the set.

iv. As the next step, all the sentences are arranged in the descending order of their \( Ws \) values. Now unmarked sentence with the highest \( Ws \) value is chosen and marked to add in the phonetically rich set of sentences.

v. Step (iv) is repeated until all unique units from the corpus are covered or all the sentences have been marked.

vi. In the next step, a set of units “\( L_u \)” is made according to some threshold value of \( Ws \). Set \( L_u \) contains the units of the sentences that have \( Ws \) values less than \( Ws \) (Threshold value of \( Ws \)).

vii. As the next step in the process, sentences from the phonetically rich sentences are manually re-structured in such a way that they can incorporate all units of the set \( L_u \). This step helped in reducing the size of the rich sentence set even further.

3. Discussion & Validation

A major benefit of this process is that it is easily adaptable to a large group of Indian Languages. This section discusses the results of this process for Hindi Language. The results for Hindi language are very encouraging and with a little manual effort it results in a set of sentences that are fairly rich and inherits the characteristics of the given text corpus. Here are a couple of sentences from phonetically rich set of sentences marked for unique syllables. This will give you a better understanding of the process, in these sentences characters in the lighter face represent the unique syllables due to which these sentence were included in the set and they are marked with 1, 2, 4, and 8 (i.e. start, middle, end and isolated respectively) which represent the position at which these syllables occurred in the word.

![Figure 2. Distribution of Syllabic words in the Text Corpus](image-url)

Figure 2 and 3 show the distribution of syllabic words in the corpus and phonetically rich set sentences, respectively.

In the process of extracting phonetically rich set from the corpus, the set is extracted in such a way that it should not loose the characteristics of the corpus. Phonetically rich set of sentences is representative of the text corpus that can be compared with the distribution of syllabic words and distribution of consonant, vowel and other symbols in both.
higher than that of the corpus, this clearly shows that the set is fairly rich as compared with the text corpus.

4. Conclusions

In this paper, we have discussed the process for automatic extraction of phonetically rich sentences from a large text corpus for Indian languages. The importance of such a system and an algorithm to generate a set of phonetically rich sentences from a large text corpus is described along with the results for Hindi language.

The future work will be focused on improving the algorithm and refining it to obtain a phonologically balanced set of sentences by further reducing the set without manual intervention and expanding the algorithm to cover sentences with complex clusters. Algorithm is also being experimented with other Indian languages like Punjabi and Marathi.

5. Acknowledgement

The work mentioned here is done under a project funded by the Ministry of Communications and Information Technology, Govt. of India. We would like to thank Dr. Om Vikas, Sr. Director, MC&IT and Shri R K Verma, Executive Director, Centre for Development of Advanced Computing, Noida and Mr V N Shukla, Director, Special Applications for constant support and conducive environment for this work. Also thanks are extended to Mr. Amit Kumar Bharadwaj for programming and to Mr. Vijay Gugnani for manual editing of phonetically rich sentences and getting the recording done.

We are grateful to Mr. C K Joseph, Controller (Administration) and Mr. R M Goel, Controller (Finance), CDAC Noida for providing necessary tools and infrastructure required in conducting research work.

6. References


