

A Note on Interpreting Text for Indian Language TTS

Kalika Bali & Amitav Das

Microsoft Research Labs India, Bangalore

kalikab@microsoft.com, amitavd@microsoft.com

1.0 Introduction

Interpreting Indic language text for Text-to-Speech Synthesis (TTS) is seen as a relatively uncomplicated task as most Indian languages have a near one-to-one mapping between orthography and sound. Though this is largely true, there are still certain issues in text processing and normalization that are specific to Indian languages and require special treatment. SSML with its aim to help TTS interpret standard orthographic text correctly in a given context, can assist in better processing of the input text in these areas and hence, enhance the output of the TTS. In this note, we would like to focus on a few specific issues which might require extension to existing SSML tag set.

2.0 Issues in text interpretation for Indian language TTS

2.1 Spelling

Indic scripts are syllabic in nature, that is, an orthographic character represents a single syllable. Thus, when a word needs to be spelt out it is usually spelt out in syllables. For example, a word in Hindi /b^har ə t / “India”, will be spelt out broken in syllables as [bha] [r ə] [t ə] . If the word is to be spelt out as phonemes, there would be no indication that the first long vowel is written as vowel ligature on the consonant and not as an independent vowel. If the word is to be spelt out in alphabets, it would require a lengthy description of all the ligatures. Thus, /b^har ə t / would require a complete description as “b^hə with the vowel *matra* for /a/, r ə , t ə ” . This can become even more complicated with words containing conjuncts and series of special ligatures.

2.2 Transliteration Schemes

By and large, phonetization of most Indic language text is not as complicated as languages like Chinese, for example, and is mostly pronounced as it is written. However, there are certain cases, in Tamil for instance, where a single grapheme is mapped to several phonemes. A TTS should be able to discriminate between the alternate pronunciations either through phonotactic rules or lexicons. In reality, creating an exhaustive lexicon of exceptions is not an easy task. Thus, being able to provide a phonetic transliteration of the input text is useful. Though IPA alphabet or other specific phonetic alphabets like SAMPA would indeed be ideal in such cases, these, especially IPA, require special training as most developers are not familiar with these. In the Indian context, certain transliteration schemes like ITRANS are in common use and almost a standard. Support for these widely accepted transliteration schemes would provide an easy way to input phonetized text in Indian languages.

2.3 Multilingual Text Documents

In a country with 18 official languages including English, it is no surprise that multilingual documents are in common use. A single document can contain text in more than one language running together. It is fairly common for texts in Indic languages to be interspersed with English words, written in Roman alphabet. This code-mixing of distinct languages requires different engines for proper rendering and would require multilingual support in SSML.

2.4 Compound Words

Compounding is an extremely productive process in Indian languages like Hindi, where almost any two (or more) words can be clubbed together to get a compound. Compounds in Hindi require special treatment in terms of certain morphophonemic rules, like schwa-deletion. Further, it is well known that compound words differ in their prosodic structure as well. The problem is complicated by the fact that in Hindi (as in other Indian languages) compounds are not separated by either space or any other separator like a hyphen. Thus, compounds in Hindi, like /lɔks ə b ʰa/ “lower house of parliament”, need to be correctly identified and interpreted to be rendered correctly as /lɔks ə b ʰa/ and not /lɔk ə s b ʰa/. Ideally, this should be done through a compound word lexicon but again this would require either appropriately tagged corpus or special morphological analysers which are not available for these languages. Thus, SSML support for marking compound words in Indian languages would facilitate the correct interpretation of these words.

3.0 Conclusion

The aim of this note is to highlight certain issues which are commonly encountered in the development of TTS systems for Indian languages. The hope is that this would generate discussion on how these can be resolved through proper support in SSML. This would make it easier for developers with no specialized knowledge in either speech or linguistics to rapidly develop high quality voice-based applications.

References

1. W3C, “Speech Synthesis Markup Language (SSML) Version 1.0”, <http://www.w3.org/TR/speech-synthesis/>
2. Deepa. S.R., A.G. Ramakrishnan, Kalika Bali, and Partha Pratim Talukdar, "Automatic Generation of Compound Words Lexicon for Hindi Speech Synthesis", Language Resources & Evaluation Conference (LREC), 2004.
3. Kalika Bali, Partha Pratim Talukdar, N. Sridhar Krishna, A.G. Ramakrishnan, "Tools for the Development of a Hindi Speech Synthesis System", In 5th ISCA Speech Synthesis Workshop, Pittsburgh, pp.109-114, 2004.