Research Report on
Bengali NLP Engine for TTS

Country Component	Bangladesh	**Report no.**	QPR – 01
Phase no.	1.2	**Report Ref no.**	PANL10n/Admn/QPR/001

Prepared By: Firoj Alam_____ **Designation: Research Programmer**
(Name)

Project Leader: Mumit Khan_____ **Signature:** _____
(Name)

Date: April 7, 2008_____

TABLE OF CONTENTS

Title

Research Report on Bengali NLP Processor for TTS

Firoj Alam, Murtoza Habib, and Mumit Khan
Center for Research on Bangla Language Processing
BRAC University

Abstract

This report describes the Bengali NLP processor for TTS, along with the challenges faced in developing the NLP processor.

Introduction

The job of the NLP processor is to take arbitrary real text as input and convert this to a form more suitable to subsequent linguistic processing. This can be thought of as an operation where we try to bring a sense of order that present in raw text and perform couple of steps to complete the whole process. The input of the system is an arbitrary raw text and output of the system is the normalized form of text with tagged linguistic information.

Methodology

The NLP processor includes several level of work to finish the whole system. Several modules includes are as follows:

1. Text normalization.
2. Text to IPA conversion
3. Syllabification
4. Stress assignment
5. Intonation assignment
6. Sound change
7. Diphone generation
8. Integration

Currently this system is able to work on text normalization, text to IPA conversion, syllabification, and diphone generation. The other modules of the NLP processor such as stress, intonation and sound change rule are not yet done. It is assumed Bangla language do not have stress and it is also assumed if stress appear in Bangla it is usually in the first syllable of the word. So it is a matter of linguistic research of stress identification. Here we assumed that Bangla language does not have any stress. Due to lack of speech corpus

of Bangla we did not worked on Bangla Intonation pattern. We included sound change module as part of text to IPA conversion by lexicon and G2P conversion.

Text normalization is the process of normalizing non-standard form of text such as number, year, date, time, acronym and abbreviation into standard form. For example, "Mr." would sound like "mister", "1st" would sound like "first", and so on. Moreover, certain numbers have to be pronounced as individual digits or as a whole. For example, a phone number such as 88028624921 will be pronounced "eight eight oh two eight six two four nine two one", but it will be pronounce as "eight thousand eight hundred and two crores eighty six lakh twenty four thousand nine hundred and twenty one" if it is referred as a measurement such as population.

Input: প্রায় সব জায়গাতেই সব সময় ১৫/১৬ কেবি পাওয়া যায়। ৩ গিগাবাইট=১৫০০/= টাকা।

Output: প্রায় সব জায়গাতেই সব সময় পনের থেকে ষোল কেবি পাওয়া যায়। তিন গিগাবাইট পনেরশ টাকা।

Text to IPA conversion: This can be thought of as a system which takes the word-based linguistic representation and generates a phonemic or phonetic description of what is to be spoken by the subsequent waveform synthesis component. In generating this representation, we make use of a lexicon to find the pronunciation of words we know and can store, and a grapheme-to-phoneme (G2P) algorithm to guess the pronunciation of words we don't know or can't store. The output of our system appeared as a two form i.e. both Bangla and IPA. A set of rules was used to generate pronounceable form of word from written text.

Input: প্রায় সব জায়গাতেই সব সময় পনের থেকে ষোল কেবি পাওয়া যায়। তিন গিগাবাইট পনেরশ টাকা।
Output: প্রায় সব জায়গাতেই সব সময় পনেরো থেকে ষোলো কেবি পাওয়া যায়। তিন গিগাবাইট পোনেরশো টাকা। /praj ʃɔb ɟajgaṭei ʃɔb ʃɔmɔj ponero ṭʰeke ʃolo kebi paoja ɟaj । ṭin gigabait ponero ʃo taka/

Syllabification: Syllabification is the separation of a word into syllables, whether spoken or written. For presentation purposes, we used dot [.] to syllabify the words into syllable.

Input: প্রায় সব জায়গাতেই সব সময় পনেরো থেকে ষোলো কেবি পাওয়া যায়। তিন গিগাবাইট পোনেরশো টাকা। /praj ʃɔb ɟajgaṭei ʃɔb ʃɔmɔj ponero ṭʰeke ʃolo kebi paoja ɟaj । ṭin gigabait ponero ʃo taka/

Output: ʃɔb ɟaj.ga.ṭei ʃɔb ʃɔ.mɔj po.ne.ro ṭʰe.ke ʃo.lo ke.bi pao.ja ɟaj । ṭin gi.ga.bait po.ne.ro.ʃo ta.ka/

Diphone generation: Diphone generation module generates sequence of diphone from word which is then map to actual diphone into the speech database.

Results

Conclusion and future remarks

This paper reports an attempt to design and implement the NLP processor for TTS. The output of this system can use in next phase i.e. phonetic module to generate waveform from the output of the NLP processor. The current NLP processor system is now in preliminary stage where couple of things requires to be fixed in future which includes compiling several modules into single one and improving the performance of each module.

References

[1] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, *Spoken Language Processing: A Guide To Theory Algorithm And System Development.* Prentice Hall, 2001. – 0130226165
[2] Paul Taylor: *Text to Speech Synthesis.* DRAFT. 2007.