Australian Journal of Science and Technology

ISSN Number (2208-6404)
Volume 3; Issue 2; June 2019

Original Article

# Syllable-based Bengali text to speech system

**Md. Kausar Ahmed[1]*, Md. Monirul Islam[2]**

[1]Department of Computer Science and Engineering, Metropolitan University, Sylhet-3100, Bangladesh, [2]Department of Computer Science and Engineering, Atish Dipankar University of Science and Technology, Sector#15, Khantek, Uttara, Dhaka-1230, Bangladesh

**ABSTRACT**

This paper describes the procedure of developing an open source and freely available Bengali text-to-speech (TTS) system based on Bengali syllable. We explored and analyzed several areas of Bengali word and syllable. Moreover, we worked with different types of Bengali syllable by breaking them down according to our working procedure as if our designed system can provide the proper and very natural Bangla pronunciation. This thesis is based on a new approach for making Bengali TTS system more natural. This research aims to enhance the current Bangla TTS system and provide a well-developed Bangla TTS program that gives a native speech output.

**Keywords:** Sound file collection, syllable detection, syllable selection, text normalization

**Submitted:** 28-04-2019, **Accepted:** 15-05-2019, **Published:** 28-06-2019

## INTRODUCTION

Text to speech (TTS) is a technology that converts a written text into human understandable voice by applying some linguistic rules and algorithm [Figure 1]. TTS system takes computer readable text and converts into audible speech. It identifies and reads aloud what is being displayed on the screen. With a TTS application, one can listen to computer text in place of reading it. A TTS synthesizer is a computer-based system that can be able to read any text aloud that is given through standard input devices.

A syllable-based Bangla TTS synthesizer is an application that converts Bangla text into spoken Bangla word, by analyzing and processing the Bangla text using natural language processing and then converts this processed text into synthesized speech representation of the text. Here, we developed a useful Bangla TTS synthesizer in the form of a simple application that converts inputted Bangla text into synthesized Bangla speech and reads out to the user.

Within Bangladesh and India, there was a high demand for TTS systems in local languages since Bangla is the 1st language for Bangladesh and Kolkata in India. Since Bangla is the mostly used language in Bangladesh and Kolkata, Bangla TTS system was implemented to provide a new way for busy computer users and visually impaired users in Bangladesh and Kolkata to effectively use computers in their day-to-day life in their native language.

Daily agenda of a busy individual is packed with so many tasks and activities. Computer and smart device users spend a lot of time reading items on screen to do their day-to-day tasks such as checking email, reading documents, collecting information from internet, and much more. It's also important to mention that blind or visually impaired people cannot perform these tasks by themselves without assistance from others. Therefore, in today's competitive world, it is very essential to make efforts to create opportunities to these disadvantaged communities by allowing convenient access of information.[2]

At the time, we are thinking about this work, English TTS has become successful. In most of devices, the uses of English TTS have become very natural and very important for users which are a great solution for the people all over the world. The use of English TTS makes peoples' life easy to easier and especially for the blind people it is a great blessing of computer science.

Moreover, where the fact is about Bangla TTS, the number of work is not much more that has done on it yet except some

**Address for correspondence:** Md. Monirul Islam, Department of Computer Science and Engineering, Atish Dipankar University of Science and Technology, Sector# 15 khantek, Uttara, Dhaka 1230, Bangladesh. E-mail: monir.duet.cse@gmail.com


**Figure 1:** Block diagram of text-to-speech synthesis system[1]

of the work from Bangladesh and India. However, the fact is none of these projects could not gain success. Almost all of them are depend on gramophone technology. They tried to make pronunciation of Bangla word by breaking them using the gramophone technology. They did not get the full success because the output of the work is not as perfect as demanded.

## History of Bengali TTS

We are not first for working in Bengali TTS system. Some work already has done on it and some of them keep continue their work. However, their result of producing natural and normal native Bengali pronunciation is not so good that can be used for everyday use of general people. We will briefly discuss about there of them named Kotha,[3] Subachan,[4] and Bangla Vaani.[5]

Several attempts were made in the past, where different aspects of a Bangla TTS system were covered. In "building Bengali voice using Festvox," authors described about different modules (optimal text selection, grapheme-to-phoneme (G2P) conversion, and automatic segmentation tools) in detail and experiment results of the different module have shown. In "some important aspects of Bengali speech synthesis system," a significant amount of work has been done for developing Bangla TTS. Phoneme and part name (similar to diphone) are used to develop voice database and ESOLA technique used for concatenation. However, quality may suffer for lack of smoothness. In "Bengali TTS synthesis system: A novel approach for crossing literacy barrier," authors showed some practical applications with Bangla TTS system using Epoch Synchronous Non-overlap Add (ESNOLA) technique. However, performance of the output not described. In "Bangla pronunciation rules and a TTS system," author showed the pronunciation rule and phoneme to speech synthesizer using formant synthesis technique. None of them have shown the naturalness and intelligibility of the system. This work is done with multiset unit selection and unit selection technique within festival framework and performance of the intelligibility and naturalness of the system has shown.

## Kotha

BRAC developed a Bengali TTS synthesis system "KOTHA" using festival.[6] Kotha is the first open source true TTS synthesis system for Bengali. Festival is a multilingual speech synthesis system which provides general framework for building speech synthesis. The system is too big and slow. It used 4355 diphones. It takes 2 s to generate a 10 s utterance. Moreover, we cannot implement Bengali character directly in festival. They divide their system into three different major phases. They are as follows:

1. Text analysis
2. Phonetic analysis
3. Prosodic analysis.

The system developed using phonology, G2P conversion, and prosodic information in the festival framework. Since festival does not provide complete language processing support specific to various languages, so it is augmented with linguistic resources to facilitate the development of TTS systems. They propose how various language processing modules such as text normalization, G2P, intonation, and duration models can be developed and integrate within festival to develop Bangla TTS system.

## Subachan

Shahjalal University of Science and Technology developed a diphone-based Bengali TTS synthesis system named "Subachan." It uses a minimum diphone set (527) for Bengali TTS synthesis. The major modules of Subachan are as follows:

1. Normalization
2. Phonetic analysis
3. Prosodic analysis and
4. Wave synthesis.

In normalization process, they have used some modules, for example, token identification, lookup table, and expansion rules for analyzing a sentence. Using these modules, they can recognize the type of each word clearly and find out the dependency on the words in a sentence. Normalization solves ambiguity problem and increases correctness. In Phonetic analysis, they have used grapheme to phoneme rules for most of the cases but to solve the problem of O-karanto problem, they have used a small dictionary containing the pronunciation of couple of words. Moreover, finally, they have applied concatenation approach on diphone to develop the system. In ideal situation, Subachan can produce better performance.

## Bangla Vaani

Bangla Vaani is a TTS for Bangla synthesis system of Kolkata-based Bengali language that capable of synthesizing adequately intonated speech. Kolkata developed this Bengali TTS system according to ESNOLA concatenative synthesis. In concatenation synthesis, speech is generated by combining splices of pre-recorded natural speech. To take care of context dependency and information embedded in transition segments, the splices are selected such that they begin and end with comparatively steady states.

ESNOLA is a concatenative speech synthesis system which uses a new set of signal units in subphonemic level, namely part name as the smallest signal units for concatenation. The ESNOLA algorithm is developed for concatenation, regeneration, as well as for pitch and duration (prosodic) modification. The methodology of concatenation provides adequate processing for proper matching between different

segments during concatenation. Bangla Vaani features:

1. Low memory requirement and runs on Windows OS
2. Support UNICODE and ISCII input for Bangla text
3. Easy to integrate with other applications
4. Supports unlimited vocabulary with text normalization
5. Output in 16-bit pulse-code modulation format with sampling frequency 22,050 Hz.

## Our Methodology and System Structure

There are four major phases in our system named text normalization phase, syllable detection phase, syllable collection phase, and sound file collection phase. Text normalization phase includes splitting word, identifying type of each word. Syllable detection phase provides different rules of pronunciation for detecting each syllable. Syllable selection phase includes splitting syllable and searching database. Finally, collection and concatenation of syllable are performed in the sound file collection phase. Thus, we produce speech from text by concatenating syllables. Moreover, after completing these phases, we expect that our designed system will be able to perform plain Bangla speaking with naturalness.

## Normalization

In normalization phase of our system, we split each word from text and Bangla sentence will be separated into words. Associated letters will be simplified and the letters that have the same pronunciation will be replaced by one by the help of some rules. We will use some symbols including comma, high pen, and white space as a delimiter. Then, we detect the category of each word because word detection is important for removing text rules and algorithm speech ambiguity problem, expanding words, etc. Moreover, word detection is helpful to make correct pronounce of each word. Thus, we normalize our raw text by splitting words, expanding according to their types, elaborating abbreviated words, etc. There are two parts of our normalization phase. One of them is replacement and another of them is conversion. Some rules of text normalization are listed here.

## Replacement

Bengali language contains 11 vowels, 10 vowel diacritics, and 34 consonants alphabet. In those numbers of alphabet, there are some alphabets which pronunciation is very close. Hence, we decide to replace them with the most matching single one. Replacement of alphabets and vowel diacritics are shown below.

| Alphabet | Replaced by | Alphabet | Replaced by | Alphabet | Replaced by |
|---|---|---|---|---|---|
| ঈ | ই | ৎ | ত | ণ | ন |
| ী | ি | য | জ | ঙ | ং |
| ৈ | ে + ই | স, ষ | শ | ঞ | য় |
| ঊ | উ | ঋ, ৃ | রি | | |
| ূ | ু | ড়, ঢ় | র | | |

Rules and examples of replacement are given below.

| Rules | Example |
|---|---|
| ঈ will be replaced by ই | ঈগল → ইগল, ঈদ → ইদ |
| ী will be replaced by ি | জীবন → জিবন, বীজ → বিজ |
| ৈ will be replaced by ে ই | জনৈক → জনে ইক, সৈকত → সে ইকত |
| ঊ will be replaced by উ | ঊষা → উষা, ঊর্মি → উর্মি |
| ূ will be replaced by ু | দূর → দুর, দূত → দুত |
| ৎ will be replaced by ত | সৎ → সত, হঠাৎ → হঠাত |
| য will be replaced by জ | যাত্রা → জাত্রা, যদি → জদি |
| স will be replaced by শ | সাহস → শাহশ, সবুজ → শবুজ |
| ষ will be replaced by শ | ষাঁড় → শাড়, ষষ্ঠ → শষ্ঠ |
| ঋ will be replaced by রি | ঋণ → রিন, ঋষি → রিষি |
| ৃ will be replaced by রি | হৃদয় → হ্রদিয়, হৃদি → হ্রদি |
| ড় will be replaced by র | বড় → বর, পাহাড় → পাহার |
| ঢ় will be replaced by র | গাঢ় → গার, রূঢ় → রূর |
| ণ will be replaced by ন | লবণ → লবন, হরণি → হরনি |
| ঙ will be replaced by ং | রঙ → রং, ব্যাঙ → ব্যাং |
| ঞ will be replaced by য় | মঞ্চি → ময়িা, ভূঁইঞা → ভূঁইয়া |

## Conversion

In Bengali language, some word does not follow the pronunciation of spelling and their meanings are also different from each other and they are not same type of parts of speech, but they just look like same to same in writing and spelling. As a result, reading them through correct pronunciation by the help of a machine is a little bit confusing.

As an example, we can see the sentence, তারা বল খলেে that means - they play ball, where বল pronounced as বল্. On the other hand, we can see another sentence, তুমি কথা বল that means – you talk, where বল pronounced as বলে◌া. Here, the word বল of the first sentence is noun and the word বল of the second sentence is verb. Although both of the words বল are look like same in spelling, they are actually not same.

Hence, we think that sentence analysis is also needed in text normalization phase to solve ambiguity problem. Most of the ambiguity problems are arisen between noun and verb. Hence, we will detect noun and verb by n-gram model, a renowned algorithm of machine learning.

1. If a letter in middle of two vowel/consonants, then we pronounced the middle letter with diacritic of ও (ে◌া).
2. If a letter in middle of two consonants and first consonant has any vowel diacritic, then we pronounced the middle letter with diacritic of ও (ে◌া).

3. If a letter in middle of two consonants and first consonant has any vowel diacritic, the second consonants also have any vowel diacritic but (ু), then we pronounced the middle letter with diacritic of ও (ো).
4. If a word contains three letters, middle letter of two vowel/ consonants, first consonant has any vowel diacritic, second consonants also have vowel diacritic of আ (া) and ও (ো); then, there will be no change in pronounce.
5. If a word contains "ঃ," then the letter "ঃ" will be pronounced as the next letter to it.

Conversion examples:

| Word | After convert | Rules applied | Word | After convert | Rules applied |
|---|---|---|---|---|---|
| অমর | অমে ার | 1 | কাগজ | কাগে াজ | 2 |
| আদর | আদে ার | 1 | কমলা | কমে লা | 3 |
| পাগল | পাগে ল | 2 | লখিন | লখি েন | 2 |
| অবগতি | অবে াগে াতি | 1,3 | এবং | এবে ং | 1 |
| মামলা | মামলা | 4 | বাংলা | বাংলা | 4 |

## Joint Letter Simplification

1. "হ্য" will be replaced by "জ্ঝ".
   Example: সহ্য à সজ্ঝ, বাহ্যকি à বাজ্ঝকি etc.
2. If "জ্ঞ" is found first at the word, then it will replaced by "গ্য়ে" otherwise it will be replace as "গ্গ".
   Example: জ্ঞান à গ্য়োন, বজ্ঞান à বগ্গিান etc.
3. If "ক্ষ" is found first at the word, then it will replaced by "খ" otherwise it will be replace as "ক্খ".
   Example: ক্ষয় à খয়, ক্ষত à খত, পরীক্ষা à পরক্খিা etc.
4. If a word starts with join letter of "ব" such as (জ্ব, স্ব etc.), then the joint letter of "ব" will not be pronounced in that word.
   Example: জ্বালা à জালা, স্বাধীন à শাধনি etc.
5. If a word starts with join letter with "স" such as (স্ম, স্প etc.), then the joint letter with "স" will be pronounced as "ইস".
   Example: স্পর্শ à ইসপরশ, স্মতি à ইসমতি etc.
6. If a word contains join letter of "ব" such as (শ্ব, স্ব etc.) with any consonants but (ম, ল, হ), then the joint letter of "ব" will be pronounced as the previous letter of it.
   Example: বশ্বি à বশিশে ো, নঃিস্বà নসিসে ো, লম্ব à লমবে ো, জহ্বিা à জহিবা etc.
7. If a word contains join letter of "য" which called as "য-ফলা" (but not at first), then the joint letter of "য" will be pronounced as the previous letter of it.
   Example: বাক্য à বাককে ো, কাম্য à কামমে ো etc.

## Syllable Detection in Words

Syllable detection is the second phase of our working method where the syllables will be detected in words. We decide to target the monosyllables, disyllables. Polysyllables will be solved with the help of monosyllables and disyllables. We provide some rules to detect the syllables from the word. The rules are given below:

1. All one lettered words are considered as a syllable.
   Examples: কে, ও, সে, etc.
2. If a letter has no vowel diacritics and the next following letter of this also has no any vowel diacritics in words, then those two letters are considered as a syllable.
   Examples: এক, বল, etc.
3. If a letter has a vowel diacritics and the next following letter of this has no vowel diacritics in words, then those two letters are considered as a syllable.
   Examples: কাজ, নাচ, দাগ, ফুল, etc.
4. If a letter has no vowel diacritics and the next following letter of this has a vowel diacritics in words, then those first letters are considered as a syllable and the second letter will follow the above rules.
   Examples: নদী à ন+দী (first follow this rule then the first rule), কলা à ক + লা (first follow this rule then the first rule), কলাম à ক + লাম (first follow this rule then the third rule), etc.
5. If a letter has a vowel diacritics and the next following letter of this also has a vowel diacritics in words, then those first letters are considered as a syllable and the second letter will follow the above rules.
   Examples: ছলেে à ছ + লে (first follow this rule then the first rule), ছলেরো à ছ + লে + রা (first follow this rule then again follow this then the first rule),

If we detect the first line of our national anthem (আমার সে োনার বাংলা), it will be look like this:

আ + মার(4,3) সে ো + নার(5,3) বাং + লা(3,1)

## Sound File Collection

Finally, we collect all corresponding sound file for each syllable and concatenate them to produce an output speech. We need to implement advanced search algorithm to reduce complexity as our database is too large (approximate 5200 sound files). We can categorize our sound files according to vowel diacritics. Some categories are listed here.

- One lettered syllables with has no vowel diacritics (C and V type syllables):
  অ, আ, ই, উ, ক, খ, গ, ঘ, etc.
- One lettered syllables with has a vowel diacritics (CD type syllables):
  কা, কি, কু, কে, কে ো,খা, খি, খু, খে, খে ো, etc.
- Two lettered syllables where two letters have no vowel diacritics (VV, CV, VC, and CC type syllables):
  কক, খক, গক, ঘক, চক, ছক, জক, ঝক, কখ, খখ, গখ, ঘখ, চখ, ছখ, জখ, ঝখ, etc.

- Two lettered syllables where the first letter has vowel diacritics (CDV and CDC type syllables):
  কাই, খাই, গাই, ঘাই, চাই, ছাই, যাই, ঝাই,কাউ, খাউ, গাউ,

ঘাউ, চাউ, ছাউ, জাউ, ঝাউ,কাক, ককি, কুক ককে, কোক, ক্যাক, কাখ, কখি, কুখ, কখে, কোখ, etc.

In our database, a huge number of syllables are CDC type, where the first letter has "কার." Hence, we can categorize them according to different "কার" for reducing searching complexity. Some categories are listed here.

- CDC type syllable where the first letter has diacritics of "আ"(া)
  কাক খাক গাক ঘাক চাক ছাক জাক ঝাক
  কাখ কাগ কাঘ কাচ কাছ কাজ কাঝ, etc.
- CDC type syllables where the first letter has diacritics of "ই"(ি):
  ককি কখি কগি কঘি কচি কছি কজি কঝি
  খকি খগি খঘি খচি খছি খজি খঝি, etc.
- CDC type syllables where the first letter has diacritics of "উ"(ু):
  কুক কুখ কুগ কুঘ কুচ কুছ কুজ কুঝ
  খুক খুখ খুগ খুঘ কুচ খুছ খুজ খুঝ, etc.
- CDC type syllables where the first letter has diacritics of "এ"(ে):
  ককে কখে কগে কঘে কছে কজে কঝে
  খকে খখে খগে খঘে খচে খছে খজে খঝে, etc.
- CDC type syllables where the first letter has diacritics of "ও"(ো):
  কোক কোখ কোগ কোঘ কোচ কোছ কোজ কোঝ
  খোক খোখ খোগ খোঘ খোচ খোছ খোজ খোঝ, etc.

## Sound File Joining

After detection of syllables in words, the next job is to join the syllables. For that, we have to careful about the cutting points of syllables. If we assume, a disyllabic word "AB" consists of "A" and "B." For pronunciation of the word, we have concatenated those syllables. The concatenation of sound example is given in Figure 2.

## Output Processing

Output is the most important things of system to judge. Output reflects the accuracy and behavior. It tells you percentage of success. We pick an article cutting from different popular newspaper and books and find the accuracy percentage of our system.

1. উপকূলে আজ শনিবার দুপুরে আঘাত হেনেছে ঘূর্ণিঝড় "রোয়ানু"। চট্টগ্রাম বিমানবন্দর ও বন্দররে কার্যকরম বন্ধ রয়েছে। সন্দ্বীপ ও বাঁশখালী উপজলোয় বড়েিবাঁধ ভঙেে লোকালয়ে পানি ঢুকে পড়ছে। বড়েিবাঁধ ভঙেে কুতুবদয়িার চারটি, মহশেখালীর তনিটি, টকেনাফরে একটি, পকেুয়ার তনিটি ইউনয়িন লন্ডভন্ড হয়ছে। কক্সবাজারে বভিন্নি এলাকার কয়কে হাজার ঘরবাড়ি প্লাবতি হয়ছে। বরগুনায় দড়ে লাখরে বশেি মানুষ পানবিন্দী হয়ে পড়ছে।[8]

Output: উ+পো+কু+লে+ আজ+ শ+নি+বার+ দু+পু+রে+ আ+ঘাত+ হে+নে+ছে+ ঘূর+ণি+ঝো়+ রো+য়া+নু+ চট+টগ+রাম+ বি+মা+নো+বো·ন+দর+ ও+ বন+দ+ররে+ কার+রক+রম+ বন+ধ+র+য়ে+ছে+ ।+ শন+দ+দপি+ ও+ বা+শো+খা+লি+

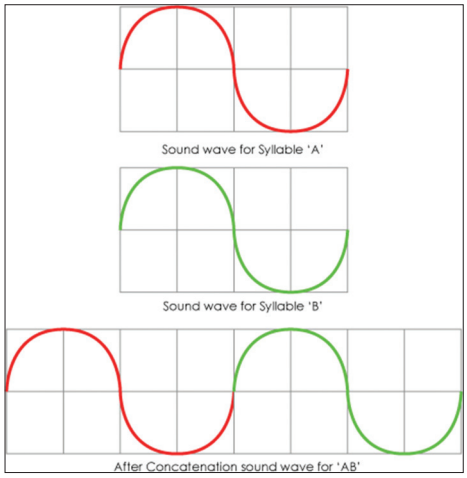


**Figure 2:** Concatenation of two sound file

উ+পো+জে+লায়+বে+ড়ি+বাধ+ ভে+ংে+ লো+কা+লো+য়ে+ পা+নি+ ঢু+কে+প+ড়ে+ছে+ বে+ড়ি+বাধ+ ভে+ংে+ কু+তু+বো+দি +য়ার+চা+রো+টি+,+ব+রো+গু+নায়+ দড়ে+ লা+খরে+ বে+শি+ মা+নুশ+ পা+নি+বো·ন+দি+ হ+য়ে+ প+ড়ে+ছে+ ।+

## Future Research Scope

Now, the system has only the main structure, but it has many scope to develop in future. Many of them are pointed below:

- Graphical user interface develop: User interface can be developed further. Because the system has only plain coding in it.
- Less concatenation: We have to bear in mind that concatenation of less sound file makes the pronunciation much smoother. Hence, less concatenation can be developed for the better pronunciation.
- DSP: PSOLA[7] can use for modifying the pitch and duration of the syllables.
- Numbers, dates, and times: There is no acceptance of numbers, dates, and times in our system. It is very important for a complete TTS system.
- Less time consumption: Time consumption between two syllables can be reduced through better DSP.
- More accuracy on detecting syllables: Although it has above 80% accuracy on detection of syllables, it can be developed for more accuracy.
- Decrease the number of syllables: We have approximately 5200 syllables for our system. Hence, we need 5200 sound files for that and it's a big number. For that reduction of syllable can make the system to perform faster.

# CONCLUSION

The architectural design of our system is completely new approach for a Bengali TTS. Satisfactory outcomes are came from our system. Finally, we can say that performance of our TTS system is better than the others. Moreover, the naturalness of pronunciation is quite high. However, it also has some

boundaries. If we cover those boundaries in future, it can be a complete system. Moreover, more analysis can make it a much accurate system.

## REFERENCES


1. Rashid MM, Hussain A, Rahman MS. Text normalization and diphone preparation for Bangla speech synthesis. J Multimed 2010;5:551-9.
2. Bandara WM, Bulathsinghala SV, Lakmal WM, Liyanagama TD. Sinhala Text to Speech for Unicode. Sri Lanka: University of Moratuwa, Faculty of Engineering; 2009.
3. Alam F, Nath PK, Khan M. Text to Speech for Bengali Language Using Festival. Available from: http://dspace.bracu.ac.bd/xmlui/handle/10361/675.
4. Naser A, Aich D, Amin R. Architectural Design of Bengali Text to Speech Synthesis Software" with Sentence Analysis using Advanced Linguistic Processing Modules: Stemming, Phrase Analysis and Expansion Rules. Sylhet, Bangladesh: CERIE; 2010.
5. ESNOLA Based Bangla TTS. Available from: http://www.cdac.in/index.aspx?id=mc_st_TTS_Bangla.
6. Festival Speech Synthesis, Speech Tools and Documentation. Available from: http://www.festival.org.
7. Available from: http://www.prothomalo.com/bangladesh/article/865249/উপকূলে-'রোয়ানুর'-আঘাত.
8. Available from: http:/en.wikipedia.org/wiki/PSOLA