International Journal of Computer Science and Applications,
©Technomathematics Research Foundation
Vol. 14, No. 2, pp. 1 – 19, 2017

REVIEW OF SYLLABLE BASED TEXT TO SPEECH SYSTEMS: STRATEGIES FOR ENHANCING NATURALNESS FOR DEVANAGARI LANGUAGES

NILESH B. FAL DESSAI

*Department of Computer Science and Technology,
Goa University, Taleigao Plateau, Goa – 403206, India
nfd@gec.ac.in
http://www.unigoa.ac.in*

GAURAV A. NAIK

*Info Tech Corporation of Goa Ltd.
Altinho, Panaji-Goa, 403001, India
gauravnaik15@gmail.com
http://www.infotechgoa.com*

JYOTI D. PAWAR

*Department of Computer Science and Technology,
Goa University, Taleigao Plateau, Goa – 403206, India
jdp@unigoa.ac.in
http://www.unigoa.ac.in*

Text to Speech (TTS) Synthesizer is an application that converts text to speech. A TTS has applications that make computer systems interactive and help its users especially the visually challenged. Concatenative synthesis technique uses different units of speech such as words, syllables, diaphones and phonemes. Most of the Indian languages are syllabic in nature and thus syllables are best suited as the unit of synthesis over phonemes and diphones. Syllabification is used in Speech Synthesis Systems in producing natural sounding speech and in speech recognizers in detecting words. This paper investigates the use of some candidate speech synthesizers inclusive of Indian and non-Indian languages in the context of high quality TTS Systems using syllable as the unit for synthesis. Analysis indicates that the syllable based approach performs better for Indian languages. Forward and backward approach to generate syllables is implemented for Indian languages written in Devanagari Script like Hindi, Marathi and Konkani for the development of a TTS system. The system is developed for different units of speech such as words, syllables, diphones and phonemes. The test results indicate positive results in terms of naturalness for Konkani language taking into account the implementation of the Konkani phonological rules. Each language has specific phonological rules requiring attention for the development of a TTS system with high naturalness and intelligibility rather than only using syllable as the unit for synthesis. Thus it becomes complex to develop a common generic TTS system for different languages.

Keywords: Speech synthesis, Syllable, Syllabification, TTS, Devanagari, Konkani

1. Introduction

The most dynamic form of communication in everyday life is speech. Speech synthesis system converts written text to speech. To build a natural sounding speech synthesis system, it is essential that the text processing component produce an appropriate sequence of units. The quality of synthetic speech produced by synthesizer in terms of naturalness is with the use of concatenative synthesis technique. Concatenative synthesis maintains a waveform repository of basic speech units that encapsulate the sounds in a particular language along with co-articulation, prosody and transitions exhibited by the system [Rao *et al*., (2005)]. One of the most important aspects of concatenative synthesis is to find the correct unit length representing the language being modeled. The basic types of units are phonemes, diphones, syllable and polysyllables [Bellur *et al*., (2011)]. As Indian languages are syllabic, syllables are best suited as the basic units over phonemes and diphones in this context. Also the concatenation points are reduced and the syllable boundaries are characterized by low energy regions with syllable as the basic unit [Kaur and Singh, (2013)]. With the development of TTS systems, researchers have resolved the environmental barriers for people with a wide range of disabilities. In our study, we attempt to analyze the techniques and strategies used by some syllabic speech synthesizers.

Various speech synthesis methods are in use today to improve the speech quality and naturalness; a few of them are reviewed here. In syllable based speech synthesis, while forming a new word, the position of the syllable is considered inorder to apply proper rules of concatenation to generate a more natural output.

The organization of the paper is as follows. Section 2 gives the overview of the syllable based TTS systems under our study. Section 3 deals with the comparative analysis of a syllable based TTS systems. Section 4 discusses the syllabification algorithms for Devanagari languages followed by the implementation details along with the evaluation of the system. Konkani phonology rules affecting syllabification along with its implementation and evaluation are discussed in section 5 followed by conclusion.

2. Overview of syllable based TTS systems

Syllabification algorithms have been proposed for various languages. According to the researchers and philologists in the domain of Indian languages, all Indian languages are phonetic in nature and thus they require a common syllabification algorithm. But practically, we find that each language has its own uniqueness and thus leading to independent work being carried out for the respective language. Some of the recently implemented synthesizers for Indian languages as well as two non-Indian languages were examined to draw comparative inferences.

2.1. *Development of an Arabic TTS system*

This speech synthesizer is a combination of formant and concatenation techniques [Zeki *et al*., (2010)]. The design of synthesizer is split into three phases: Pre-Processing, Natural Language Processing and Digital Signal Processing. Pre-Processing phase prepares raw text for processing i.e. it detects spaces, punctuation marks or non-Arabic symbols in each word of the sentence. The Natural Language Processing phase performs mapping of each word to its exact phonetic representation in three parts. Words are searched in the exception dictionary containing list of all words whose pronunciations are explicitly given rather than determined by the pronunciation rules in the first part. In the second part, Arabic pronunciation rules are applied. These rules specify phonemes which are used to pronounce each letter or sequence of letters. To find the pronunciation of the word, the rules are searched with the highest score depending on how many letters are matched. In the third part, all phonemes are defined in Arabic language. Words are applied with special patterns for syllable lexical stress.

Finally in the Digital Signal Processing phase, the resultant phonemic representation of input text with special stress is transformed into a proper utterance wave file.

2.2. *Indonesian TTS system using syllable concatenation for PC-based low vision aid*

This TTS synthesizer uses syllable concatenation [Soedirdjo *et al.*, (2011)]. The list of all syllables required to make all words in Bahasa (Indonesian national language) are recorded with the same characteristics. The input special terms are handled by exception library and numeric characters are normalized to their sounds. The input word is seen as a list of characters wherein, the word is processed to obtain the proper combination of syllable i.e. first the system will check whether there are reserved syllable patterns for the particular word. If reserved pattern is available, then the system will use that pattern. Otherwise, the system uses brute force algorithm to obtain the syllables required to form the word. This is followed by matching the word by its longest pattern and checking for its corresponding sound in the library. If no matching sound is available, then check for shorter pattern. This algorithm is repeated until the word is processed.

In this synthesizer prosody is generated by looking for punctuation at the end of the sentence. In this study, we concentrate on two general punctuation that are commonly used: dot (.) and question mark (?). If a sentence ends by a dot (.), then the pitch of the syllable before the mark is lowered by a semi tone. If it ends by a question mark (?), then the pitch is raised by a semi tone.

2.3. *Marathi TTS system using concatenative synthesis strategy*

This paper presents a concatenative speech synthesis technique for Marathi language (Spoken in Maharashtra, India) using different choice of units: words, syllables and phonemes [Shirbahadurkar and Bormane, (2009)]. The text input is either a standard word or a non-standard word. If the input text is a number then it is handled by a digit processor. The TTS system is able to read any dates, addresses, telephone numbers and bank account numbers. If input text is a word then it searched in the word database. If the word does not exist in the database then it is cut into syllables and syllables are searched in the syllable database. If the corresponding syllable does not exist in the database then the word is formed by concatenating phonemes from the phoneme database and played.

2.4. *TTS system for Punjabi language*

This paper implements a text to speech synthesis system for the Punjabi text written in Gurmukhi script (Punjabi Language) [Singh and Lehal, (2006)].

Concatenative method has been used to develop this TTS system using syllables as the basic unit of concatenation. The system is based on Punjabi speech database that contains the starting and ending position of the syllable sounds labelled carefully in a wave file of recorded words. The input text is first processed and then syllabified with an automatic syllabification algorithm that is developed based on grammatical rules for Punjabi language. These syllables are then searched in the database for corresponding syllable sound positions in recorded wave file. The quality of the output speech depends on how carefully the speech units are labelled in the recorded sound file.

2.5. *TTS system for Tamil language*

This system uses a concatenative based approach to synthesis wherein, concatenation happens at the word level and syllable level [Sangeetha *et al.*, (2013)]. The following stages are used to convert Tamil text to speech: text normalization, sentence splitting, speech corpus, concatenation and speech synthesis.

In text normalization, punctuations such as double quotes, full stop, comma, etc. are all eliminated to obtain plain sentences. In sentence splitting, the given paragraph is split into sentences using white spaces as delimiter. The quality of synthesized speech waveform depends on the number of realization of various units present in the speech corpus. The concatenation of speech files is carried out as the final stage process in MATLAB. The speech synthesis follows two approaches. First, word level synthesis; when all the words in the input text are already present in the speech corpus and hence the synthesized output naturalness is high. Second, syllable level synthesis; when the input word is not present in the speech corpus, the word is synthesized using syllable level concatenation and hence naturalness drops in comparison to word level synthesis.

2.6. *TTS system for Telugu language*

This system uses concatenative method of synthesis with syllables as the basic unit [Kumar *et al.*, (2014)]. The process of speech synthesis starts with text processing where the input Telugu text is converted to unicode irrespective of the platform and size. This is followed by differentiating Graphemes and phonemes in order to provide the correct output speech signal to the user. Phonemes can be further differentiated on the basis of CV structure (consonant & vowel). The input text it is made to go through the process of prosodic modelling and acoustic synthesis so as to generate the correct audible accent for the output speech. Selection of a syllable unit is carried out on the basis of preceding and succeeding context of the syllables and the position of the syllable.

3. Comparative Analysis of Syllable Based TTS Systems

In general the TTS systems use a variety models, techniques and tools to suite their requirements. In our work for the purpose of comparison we have considered systems using similar synthesis techniques for development. The basis for comparison considered is the nature of the script, syllable pattern, prosody modelling. The reviewed speech synthesizers are examined to draw comparative inferences considering the following parameters of a syllabic speech synthesizer: Front end processing, Symbolic prosody control, Syllable patterns distribution, grapheme to phoneme conversion [Repe, (2010)].

3.1. *Front end processing*

It refers to the first step wherein raw text (such as non-standard words) requiring normalization are transformed into pronounceable words [Naser *et al*., (2010)]. After tokenizing and token identification, expansion rules are applied to resolve ambiguity and phrase detection.

3.2. *Symbolic prosody control*

Prosody is also referred to as supra-segmental features and deals with two major factors:

- Breaking the sentence into prosodic phrases, possibly separated by pauses.
- Assigning labels, such as emphasis to different syllables or words within each prosodic phrase.

Words are normally spoken continuously, unless there is a specific linguistic reason to signal discontinuity. The term juncture refers to prosodic phrasing i.e. where words cohere, and where prosody breaks (pauses and/or special pitch movements occur).

3.3. *Syllable patterns distribution*

Syllabification is a process by which sound units are produced corresponding to the syllables. It is based on the syllable distribution patterns in the language. Generally, linguists define the syllable structure as C*VC*, where C is consonant and V is vowel. In other words, the syllabic pattern can be of the form zero or more consonants followed by a vowel further followed by zero or more consonants. Common supportive syllable patterns for Indian languages are V, CV, CCV, VC, and CVC. Given general patterns of syllable, they exhibit differently based on their language origin.

3.4. *Grapheme to phoneme conversion*

There are three approaches available for grapheme to phoneme conversion namely, dictionary based, rule based and lexical accent based [Repe, (2010)]. In the dictionary based approach, a large dictionary containing all the words of a language and their correct pronunciations are stored in the database. Although this approach is quick and accurate, it fails if look- up word is not present in the dictionary. In the rule based approach, pronunciation rules are applied to the words to determine their pronunciations based on their spellings. In the lexical accent approach, the type of pronunciation or sound of a word depends on its context.

In syllable based speech synthesis, while forming a new word, the position of the syllable is considered in-order to apply proper rules of concatenation to generate a more natural output. A brief comparison of the speech synthesis techniques for various TTS systems outlined in the preceding section is shown in Table 1. It is observed that most Indian languages use syllable based approach to speech synthesis.

Table 1. Comparison of Syllable Based Candidate TTS Systems

Parameter	**TTS systems**					
	Arabic Speech Synthesizer	***Indonesian Speech Synthesizer***	***Marathi Speech Synthesizer***	***Punjabi Speech Synthesizer***	***Tamil Speech Synthesizer***	***Telugu Speech Synthesizer***
Languages Supported	Arabic	Bahasa	Marathi	Punjabi (Gurmukhi)	Tamil	Telugu
Front-End Processing	Detects spaces, punctuations and other non-Arabic symbols	Detects units of physics, length, mass, etc. Handled by Exceptional Library.	Detects numbers, dates, address, telephone nos. & bank account nos.	abbreviations, numeric values, special symbols are analyzed	remove all types of punctuations and process pure sentences	No specific pre-processing are considered
Type of Approach	Rule Based with Dictionary Based supportive	Dictionary Based	Rule Based	Rule Based	Rule Based	Rule Based
Choice of unit for Synthesis	Syllables with words	Syllables	Syllable with words and phonemes	Syllables	Words and Syllables	Syllables
Syllable Patterns	CV, CVV, CVC, CVCC, CVVC, CVVCC	V, VC, VCC, CV, CVC, CVCC, CCV, CCVC, CCVCC, CCCV, CCCVC	C, V, CV, CCV, VC, CVC	V, VC,CV, VCC, CVC, CCVC, CVCC	V, VC,CV, VCC, CVC, CCVC, CVCC, CC, CCV	C, V, CV, CCV, CVC
Prosody	Performs lexical stress at syllables	Pitch control by semi tone raised/lowered for punctuations	No specific prosodic features are considered	No specific prosodic features are considered	No specific prosodic features are considered	Prosody considered while implementing

4. Syllabification Algorithms for Devanagari Languages

Devanagari is an alpha-syllabary script and is widely in use across India and Nepal. It is written from left to right, has a strong preference for symmetrical rounded shapes within squared outlines and is uniquely identified by a horizontal line that runs over the letters. The Devanagari script is used for over 120 languages and dialects. E.g. Sanskrit, Hindi, Marathi, Nepali, Pali, Konkani, Bodo, Sindhi, Maithili etc. Thus Devanagari is one of the most used and adopted writing systems in India [Devanagari script].

For a TTS synthesizer, the basic unit of synthesis is phoneme, diphone, syllable, word, phrase or even sentence. Table 2 outlines a brief comparison of the unit of synthesis with respect to concatenation points, database size and quality of speech generated. It is observed that with the increase in the unit size there is an increase in the database size, decrease in the concatenation points and improvement in the produced speech quality.

Table 2. Comparison of units of synthesis

Parameters	**Unit of Synthesis**		
	Phonemes/ Diphones	***Syllables***	***Words***
Concatenation points	Large	Moderate	Few
Database size	Small	Moderate	Large
Quality of Speech Generated	Good	Better	Excellent

Based on the study of the syllable based TTS systems, we propose a combined top-down approach for developing a TTS system using different choice of units: words (recorded), syllables (recorded) and syllables (formed using syllabification algorithm as shown in Fig. 1.

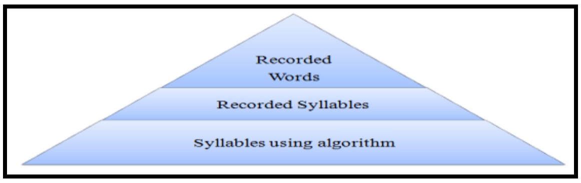

Fig. 1. Top-down flow to play input text.

The input word is first searched in the database and played if present. If the word does not exist in the database then it is cut into syllables then syllables are searched in the recorded syllable database and played if present. If the corresponding syllable does not exist in the database then syllable is formed using syllabification algorithm by concatenating phonemes/diphones from the database and played as shown in Fig. 2.

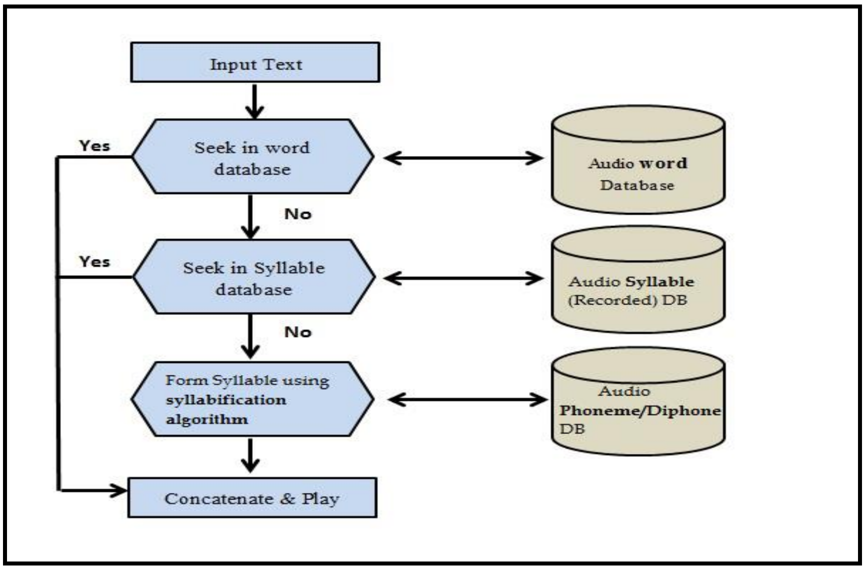

Fig. 2. Flowchart for speech generation for input text.

4.1. *Syllable Generation*

In the preceding sections, we discussed some of the TTS systems in the context of Indian and non-Indian languages. It is observed that in Indian languages there are common patterns of syllables. Syllables can be formed scanning words either from left to right or from right to left. Most of the languages are written / spoken from left to right and literature speaks of the forward approach and backward approach. Also study of few other TTS algorithms reveal that better syllabification can be obtain through the backward approach, even though that language is written and spoken form left to right [Chaudhury M].

4.2. *The Backward Approach*

The backward syllabification algorithm can be well understood in three phases. In phase I, it scans each token from right to left i.e. from rightmost character to the leftmost character (Refer Fig. 3). If the character to be traversed is ‘C’, i.e. consonant, then it follows to the phase II (Refer Fig. 4). On encounter of specific symbols, it increments the counter of symbols scanned. Phase II can identify patterns of type CC, CVC, VC and C. Similarly if the character to be traversed is ‘V’, i.e. vowel, then it follows to the phase III (Refer Fig. 5). On encounter of specific symbols, it increments the counter of symbols scanned. Phase III can identify patterns of type CV and V.

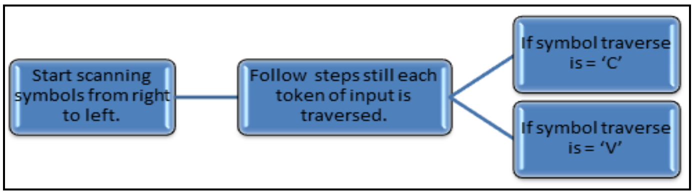
Start scanning symbols from right to left.
Follow steps still each token of input is traversed.
If symbol traverse is = ‘C’
If symbol traverse is = ‘V’

Fig. 3. Backward Syllabification Algorithm Phase I

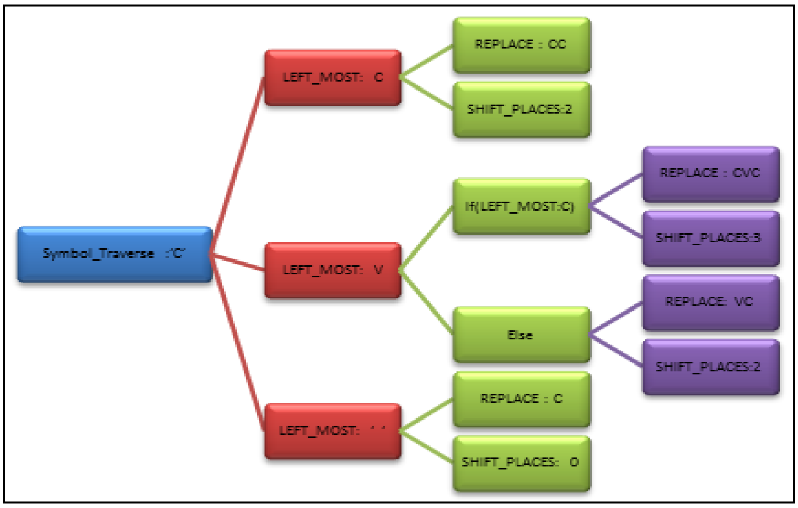

Fig. 4. Backward Syllabification Algorithm Phase II

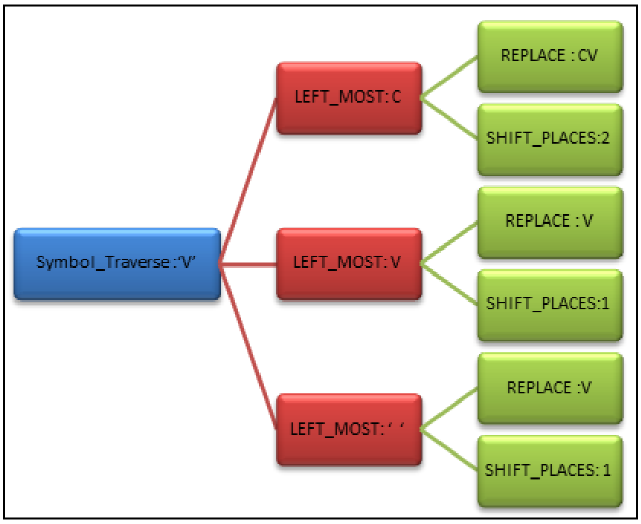

Fig. 5. Backward Syllabification Algorithm Phase III

The syllabification approach recognizes syllable patterns of the form V, C, CV, VC, CVC and CC for Devanagari Konkani. Thus the output will be string of Cs &Vs. The syllable of type 'C' and 'CC' are having presence of inherent schwa present with each consonant [Goibab, (1940)].

4.3. *The Forward Approach*

In order to compare and implement the system for forward approach of syllabification, we process the all the steps to those of backward tracing, that is we scan the characters from left to right. The algorithm can be easily understood by flow diagram shown in Fig. 6, 7 and 8 respectively.

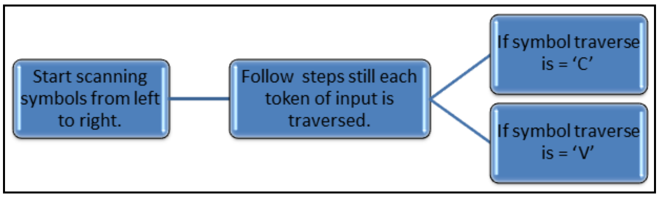
Start scanning symbols from left to right.
Follow steps still each token of input is traversed.
If symbol traverse is = 'C'
If symbol traverse is = 'V'

Fig. 6. Forward Syllabification Algorithm Phase I

4.4. *Sound inventory*

Considering 36 consonants and 12 vowels as the basic units (phones and diphones), 36+12=48 phones and 36*12= 432 diphones can be generated as part of the TTS Corpus [Goibab, (1940)]. In view of our specific requirement and the non-availability of audio sounds for syllables, for our implementation, we recorded a speech of around 500 Devanagari Konkani sentences in a studio environment. These sentences were cut into words and further into syllables of which we used around 1000 words and 1000 syllables for testing. We also recorded phonemes and numbers (0 to 100) as part of the developed speech corpus. In addition, specific syllables were recorded for the purpose of testing. The Audacity tool [Audacity tool] was used for recording, editing and analyzing which is an open source tool, easy-to-use and multilingual audio editor and recorder for Windows, Mac OS X, GNU/Linux and other operating systems. The WaveSurfer tool [Wavesurfer tool] was used to cut and label the wave files and is an open source tool for sound visualization and manipulation.

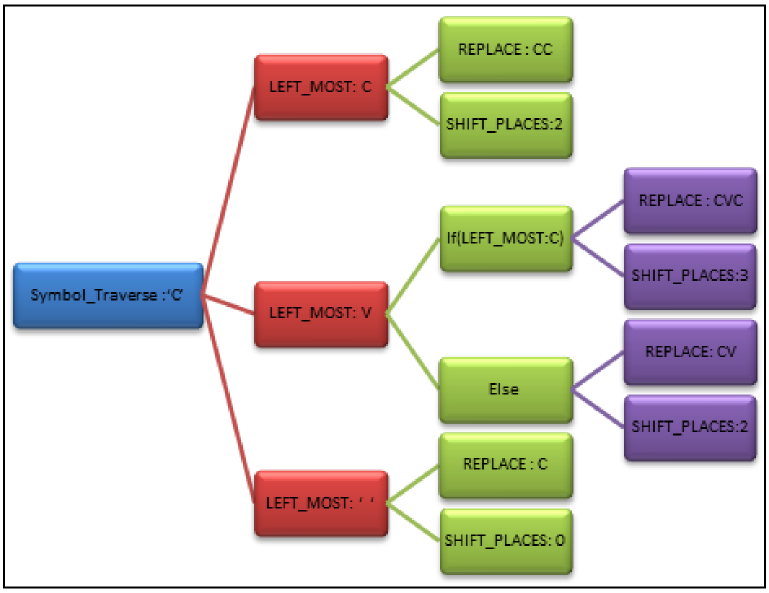

Fig. 7. Forward Syllabification Algorithm Phase II

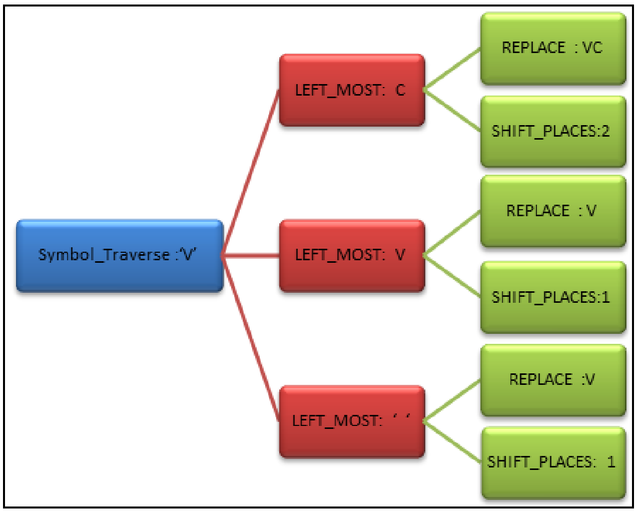

Fig. 8. Forward Syllabification Algorithm Phase III

The ultimate goal is to have a large number of syllables from the available text. An ideal text was selected for recoding the speech corpus keeping in mind the minimal presence of large polysyllables. For our work, we have used the Konkani text from the Linguistics Data Consortium for Indian Languages (LDC-IL), MDRD, Government of India [Linguistics Data Consortium for Indian Languages]. The text used is in pure Konkani language with the presence of negligible non-Devanagari text and non-standard words [Sadeque *et al*., (2013)].

The syllables present in the words can be identified and processed either through a manual or an automatic process. The cutting of the word into syllables must be very accurate to ensure correct results. A tool called SOUND FORG is used to automatically form regions according to the settings and each region gives the syllables present in the word. The length of the syllable, their start and end location which is required for concatenation can be obtained using the view region list option [Shirbahadurkar and Bormane, (2009)]. In our implementation we have used the manual process of cutting the syllables to ensure accuracy.

4.5. *Data Sets used*

For our experiments we have used Devanagari Konkani text data available online at the LDC-IL, Mysore [Linguistics Data Consortium for Indian Languages] as well as short Konkani stories provided by Konkani linguists. With the speech recording of this data, sentences, words, syllables, diphones and phonemes were generated using tools in addition to recording basic sounds and numbers for konkani. A sample text file to generate speech is as shown below:

मराठी चार पुस्तकां जातकच एकतर घरा बसून शिवण सूत करप वा घर संवसार शिकप.
व्हेल्ल्या घरा त्रास जावचे न्हय म्हूण तेन्नाच्यो आवयो आपल्या चलयांक घर संवसाराचीं कामां मुद्दाम शिकयताल्यो.

Table 3. Syllabified Words with Syllabification Algorithms

Devanagari Language	Sentence	CV Structure	Syllables Generated using Forward Approach	Syllables Generated using Backward Approach
Hindi	**वहा शायद कुछ पताचले।** (*Vaha Shayad Kuch Patachale*)	CCV - CVCC - CVC - CCVCCV	वह ा – शाय द – कुछ - पत ाच ले	व हा – शा यद – कुछ - प ताच ले
Marathi	**बाघारू ठेच लागून पडतो.** (*Bhagharu Thech Laagun Padto*)	CVCVCV - CVC - CVCVC - CCCV	बाघ ार ू – ठेच – लाग ून - पड तो	बा घा रू – ठेच – ला गून - पड तो
Konkani	**रमाबाय पणजे शारात वाडिल्ली.** (*Ramabai Panaje Shaarat Vadilee*)	CCVCVC- CCCV- CVCVC- CVCVCCV	रम ाब ाय – पण जे – शार ात – वाड ि‌ल ली	र मा बाय – पण जे – शा रात – वा डिल ली

4.6. *Experimental Results and Evaluation*

We have implemented both the forward and backward syllabification approach for three Devanagari languages; Hindi, Marathi and Konkani for the generation of syllables for words in the given input text. Table 3 depicts sample sentences with syllabified text. The in-correct syllabified words are shown with highlighted background. The simulation results show 100% accuracy using the backward approach for syllabification.

Table 4 shows the hit count for recorded words, recorded syllables, recorded diphones and phonemes for sample Hindi, Marathi and Konkani sentences presented to the developed syllabification system using the backward approach.

Table 4. Sample Sentences Tested for the Backward Syllabification

Devanagari Language	**Sentence**	**Syllabification (Backward tracing)**		
		No. of hits for recorded words	***No. of hits for recorded syllables***	***No. of hits for diphones/ phonemes***
Hindi	**आज-कल तो स्कूलो की छुट्टी है ।** (*Aaaz-Kal To Scholo Ki Chutti Hai*)	**2**	**2**	**4**
Marathi	**पण पुढच्याच क्षणी आणखी एका धक्क्यात बहादुर गाडी पुढे नेतो.** (*Pan Phudchya kshani Aankhi Eka Dhakkyat Bhahadur Gaddi Pudhe Neto*)	**3**	**4**	**5**
Konkani	**विनोदाचीं अशे तरेची मता जावपाक एक कारण आशिल्ले.** (*Vinodache Ashe Tareche Mata Javapak Ek Karan Ashelee*).	**2**	**4**	**6**

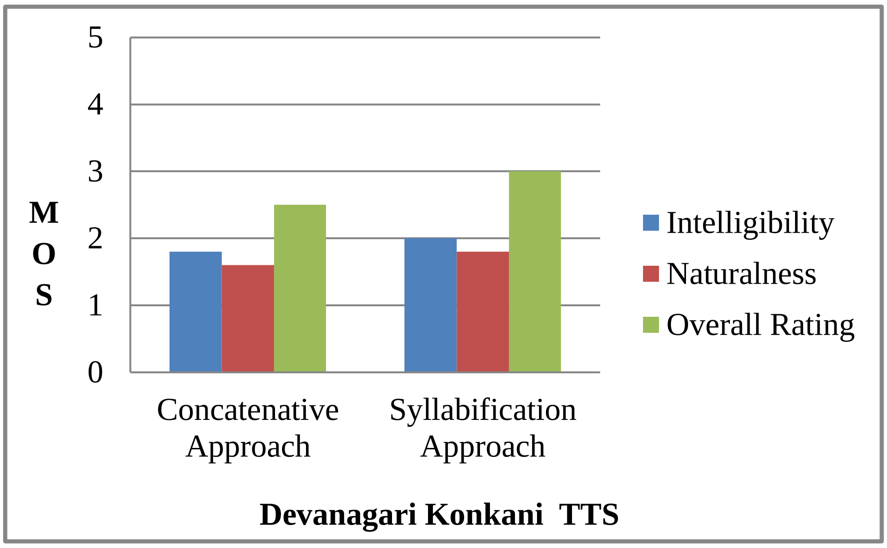

Fig. 9. Feedback of the Volunteers for the TTS System using Concatenative and Syllabification Approach

The performance evaluation of the developed TTS system was carried out using a volunteer based evaluation. Ten native male and female speakers each of Konkani in the age group of 20 to 25 years who did not have any prior exposure to speech synthesis experiments were made to listen to the speech for parameters like intelligibility, naturalness and overall rating and give their score ratings on a scale from 1 to 5 {1-Bad, 2-Poor, 3- Fair, 4-Good, 5-Excellent}. Using these scores, the Mean Opinion Score (MOS) is the average of the scores given by volunteers for a particular parameter for the speech under evaluation [Loizou].

We have used the MOS to compare the speech quality of the TTS system using concatenative synthesis [Fal Dessai *et al.*, (2016)] of our earlier work with the developed TTS system using backward syllabification approach for Konkani. The feedback of the volunteers for the two TTS systems is depicted in Fig. 9.

Based on the outcome of the listeners test, it is observed that the developed system using syllabification shows overall improvement for Konkani text for synthesis. On analyzing the extensive tests carried out by exposing the system to more Konkani text considering Konkani phonology [Devnagari script] [Tadkodkar and Mandgutkar] [Dhume] and listener's feedback, following are the observations:

- Recorded Syllables are best suited for generation of speech as compared to the speech obtained from syllables formed by concatenation diphones/ phonemes.
- Syllables formed using diphones/phonemes do not differentiate the presence of inherent schwa in consonants. For example the word, 'घर' (ghar) after deletion of the inherent schwa should be read as 'घर्' (ghar) which is actually spoken as 'घर'(ghar**a**) due to the uttering of the consonants straight forward from the database without any processing.
- Syllables formed using diphones/phonemes do not recognize or classify the words for the presence of nasal sounds represented by '◌ं' over consonant or vowel. For example in the word 'झुंबर' (jhu**m**ber), the nasal sound is represented as 'म्' (m). e.g. झुंबर -> झु+म्+बर
- Syllables formed do not differentiate the effect of vowel harmony for specific words. For example the phrase 'ती तेंफळ' uttered as 'ती तेंफळ' (tee teefal) represents the tree and the phrase 'तें तेंफळ' uttered as 'तें तॅफळ' (te tefal) represents the fruit of the tree. It is significant to note that, the word is written as 'तेंफळ' in both the phrases.
- Syllables formed using diphones/phonemes do not consider diphthong wherein sound units of type इव (Ev), इय (Ey), उव (Uv), उय (Uy), एव (Iv), एय (Iy), ओव (Ov) and ओय (Oy) are differently uttered. For example in the word 'दिवज' (divaj), there is stress on the first vowel and 'व' (V) remains silent, resulting in the utterance of sound unit 'दिवज् (इव)'
- Not adequately considered for sentence and words level stress as prosody control in synthesis.

Thus in context of the above observations, we infer that in order to get better naturalness, it is essential that the TTS system address the language specific issues.

Konkani is the official language of the State Goa in India and is written in Devanagari as well as Roman script. Very little resources are available for Devanagari Konkani in the context of phonology for the TTS systems as compared to the other Devanagari languages. Thus we have considers the above rules for Konkani language for implementing a Konkani TTS system which are discussed in the next section.

5. Konkani Phonology rules affecting Syllabification and its implementation

This section discusses phonology rules [Fal Dessai *et al.*, (2016)] for Konkani language for implementation.

5.1. *Schwa deletion rules*

In Konkani some words express different meaning depending on their position in the sentence. For such words, without schwa deletion, the word not only sounds unnatural, but also makes it extremely difficult for the listener to infer the correct sound. Thus the pronunciation of schwa is context dependent in Konkani.

The rules for schwa deletion are well known for Konkani language and are depicted in Table 5 with examples.

Table 5. Schwa deletion rules with examples.

Character Based Rules	**Utterance Rule**	**Sample Word**
Single letter words	Presence of '**अ**' (a) is always uttered	**व** (Va), **ह** (Ha), **क** (Ka)
Words terminating with '-**ना**'(Naa) or '-**नात**'(Naat)	Presence of '**अ**' (a) in the previous character is un-uttered	**वचना** → **वच्-ना** (Vach**a**na) →(Vachna) **पाव्-नात** (Pav**a**nat) →(Pavnat)
Words with **जोडगिरा** (jodhakshars)	Presence of '**अ**' (a) in the jodhakshars is uttered.	**घुस्पले** → **घुस्पले** (Ghus**a**p**a**le) →(Ghuspale)
Three letter words	Presence of '**अ**' (a) in the middle is uttered while at the end remains silent	**माजर** → **माजर्** (Majara) → (Maj**a**r)
Four letter words	Presence of '**अ**' (a) at the second position remains silent while '**अ**' (a) at the third position is uttered.	**मणकट** →**मण्-कट्** (Manakata) → (Mankat)
Five letter words	Presence of '**अ**' (a) at the second position remains silent**.** At the third position, '**अ**' (a) is uttered if present. At the last position, remains un-uttered and if '**अ**' (a) is present prior to this position then '**अ**' (a) is uttered.	**लकलकप** → **लक्-लक-प्** (Lakalakapa) → (Laklakap)

Table 6. Jodhakshars rules with examples.

Input Word	**Pre-processing Steps**	**Input String**	**Syllables Encountered**	**Syllabified Word**
अर्वळ (Arval)	**अर् – वळ**	VCCC	VC+CC	**अर् – वळ**
विठ्ठल (Vithal)	**विठ् – ठल**	CVCCC	CVC+CC	**विठ् – ठल**
माळ्ळो(Maaloo)	**माळ् –ळो**	CVCCV	CVC+CV	**माळ् – ळो**

पन्नासावी (Pannasavi)	**पन् – नासावी**	CCCVCVCV	CC+CV+CV+CV	**पन् – ना - सा – वी**

5.2. *Jodhakshar concept and its syllabification*

In Devanagari languages, two or more consonants can occur as a single unit. They are usually referred to as jodhakshars or jodgir (जोडगीर अक्षरांत). Jodhakshars and the syllabification rules for Konkani language are presented with syllabification examples in Table 6.

5.3. *Nasal utterance and its syllabification*

Nasalization is utterance of sound through nose (*नाखयो उच्चार*). Nasal sound in Devanagari is represented by '◌ं' (*अनुस्वार* **-** Anuswar) over consonant or vowel. Nasal utterance is specifically of consonants of type न् (na), *ण्* (ṇa), *ङ्* (ṅa) and *म्* (ma)**.** Hence there is a need to carry out pre-processing of words involving nasal sounds *(अनुस्वार)* followed by the normal flow of algorithm. Table 7 depicts nasalization rules with examples for Konkani. Thus, it is noted that, अनुस्वार depends on succeeding letter and hence it may be tough to predict the output when अनुस्वार is on the last letter of the word. In such cases, pre-processing is suitable.

Table 7. Nasalization rules with examples.

Nasal Utterance Character	**Preceding Character**	**Sample Word**
ङ्(ṅa)	**क, ख, ग, घ**	**रंग → र+ङ्+ग (Rang)**
न्(na)	**च, छ, ज, झ, त, थ, द, ध, न**	**हांचो → हा+न्+चो (Hancho)**
ण्(ṇa)	**ट, ठ, ड, ढ, ण**	**पंटू → प+ण्+टू (Pantu)**
म्(ma)	**प, फ, ब, भ, म**	**झुंबर → झु+म्+बर (zhumber)**

5.4. *Diphthong concept and its syllabification*

A diphthong (संदी-स्वर) is "two sounds" or "two tones" and refers to two adjacent vowel sounds occurring within the same syllable. A diphthong is a vowel wherein the tongue moves during the pronunciation of the vowel

In Konkani, vowels 'आ (Aa)', 'इ (i)', 'उ (u)', 'ए (e)' and 'ओ (Au)' when adhered to semi-vowels 'य (y)' and 'व (v)', diphthongs (संदी-स्वर) are formed. Thus we have type इव (Ev), इय (Ey), उव (Uv), उय (Uy), एव (Iv), एय (Iy), ओव (Ov) and ओय (Oy). Table 8 depicts diphthong rules with examples for Konkani.

Table 8. Dipthongs rules with examples.

Input Word	**Input String**	**Syllables Encountered**	**Diphthong**	**Syllabified Word**
कायल (Kayal)	CVCC	CVC+C	**आय**	**काय – ल**
जेवन (Jevan**)**	CVCC	CVC+C	**एव (अेव)**	**जेव – न**
रोयण (Royan**)**	CVCC	CVC+C	**ओव**	**रोय – ण**

लुवंया (Luvaya**)**	CVCCV	CVC+CV	**उव (अुव)**	**लुवं – या**
घेवंक (Ghevak**)**	CVCC	CVC+C	**एव (अेव)**	**घेवं – क**
मेवणी (Mevni**)**	CVCCV	CVC+CV	**एव (अेव)**	**मेव – णी**

5.5. *Vowel harmony and its utterance*

Vowel harmony is a type of long-distance assimilatory phonological process involving vowels that occur in some languages. In such languages, there are constraints on which vowels may be found near each other. In Konkani each of vowels 'अ', 'ए' and 'ओ' has two vowel variations. That is two 'अ', two 'ए' and two 'ओ' are important variation of vowels in its script.

Let us try to understand each of these cases with an example as shown in Table 9. If the word occurs in two different contexts then it is uttered differently. The way of writing the words may be reasonably same, but their utterance is not the same way. Aksharas can be explicitly noted by giving चंद्र (ॅ). But the use of such explicit चंद्र (ॅ) is not noted in Konkani.

Considering the above phonological rules affecting syllabification, the sound inventory was appended with the following for improving the naturalness of the produced speech:

- Consonants(in Konkani): with schwa and without schwa
- Vowels(in Konkani): including for vowel harmony for 'ए' and 'ओ' type
- Nasal, Diphthong sounds (in Konkani)

Table 9. Vowel Harmony rules with examples.

Input Phrase	**Vowel Harmony Character ('ए' and 'ओ')**	**Utterance**	**Meaning of the Context**
तें आंतेर (Te Aanter**)**	**ए**	**तें आंतॅर (**Te Aanter**)**	Represents fruit of a tree
ती आंतेर (Tee Aanteer**)**	**ए**	**ती आंतेर (**Tee Aanteer**)**	Represents tree
तें बोर (Te Bor)	**ओ**	**तें बॉर** (Te Bor)	Represents fruit of a tree
ती बोर (Tee Boor**)**	**ओ**	**ती बोर (**Tee Boor**)**	Represents tree

5.6. *Sound Inventory*

The test speech corpus created and discussed at section 5.3 was appended with phonemes, syllables and words involving phonology and text normalization for Konkani as detailed in the preceding sections.

5.7. *Data sets used*

The data sets discussed at section 5.4 was used for testing. The system was tested with a sample text of approximately 100 Konkani sentences inclusive of words containing jodakshars, nasals, diphthongs, vowel harmony and schwa deletions. Different test cases were presented, evaluated as discussed earlier.

5.8. *Experimental Results and Evaluation*

The feedback of the volunteers for the TTS systems to compare the syllabification approach and syllabification with Konkani phonology approach in terms of the Mean Opinion Score (MOS) is depicted in Fig. 10. Based on the outcome of the MOS test, it is observed that the output speech performs better when phonology features of Konkani language like jodakshars, nasals, diphthongs, vowel harmony and schwa deletions are considered for syllabification.

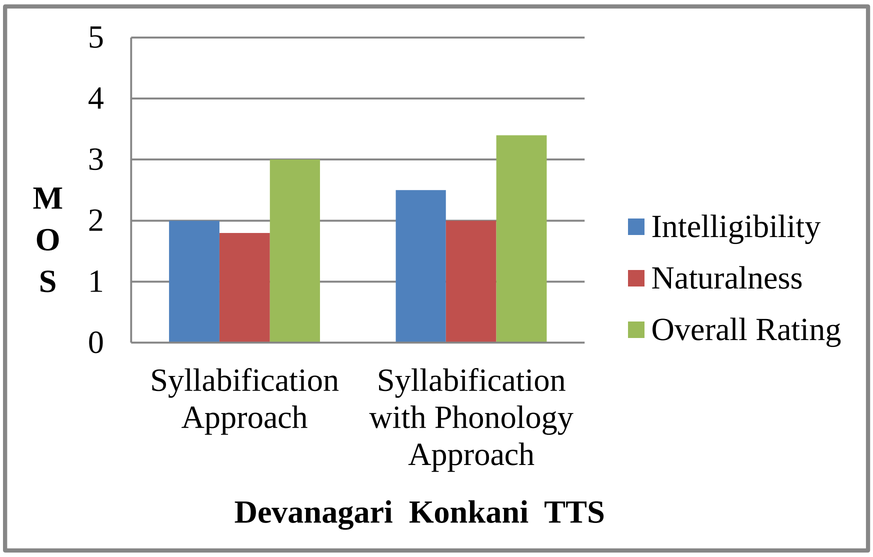

Fig. 10. Feedback of the Volunteers for the TTS System using Syllabification and Syllabification with Phonology Approach

Conclusion

Each of the strategic TTS systems studied shows its importance for the specific language for which it is designed. Syllable based concatenative approach is most suitable for Indian languages as Indian languages are phonetic in nature. Syllable formation using backward tracing is best suited for syllabification of Devanagari languages. In order to produce quality output, the synthesizer additionally requires considering text processing rather than simply using syllabification. This paper addresses few phonological rules for Konkani language like schwa deletion, jodaksharas, nasal words, diphthongs and vowel harmony along with its implementation. The speech synthesis output from the implemented system needs to be checked and compared with other available systems for performance analysis.

There is scope to study and implement more such language specific requirements to improve the naturalness and intelligibility of the TTS system.

References

Audacity Tool: http://www.audacityteam.org.

Bellur A., Narayan K. B., Raghava K. K, Murthy H. A. (2011): Prosody Modelling for Syllable-Based Concatenative Speech Synthesis of Hindi and Tamil, Proceedings, National Conference on Communications.

Chaudhury M., “Rule Based Grapheme to Phoneme Mapping for Hindi”: http://citeseerx.ist.psu.edu.

Devanagari Script: https://en.wikipedia.org/wiki/Devanagari

Dhume V. M.: Konkani Shudhlekanache Nhem, Goa Konkani Academy, pp 6-11.

Fal Dessai Nilesh B., Naik Gaurav A., Pawar Jyoti D. (2016): Development of Konkani TTS System using Concatenative Synthesis, Proceedings of the IEEE International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kanyakumari, PP. 425- 429

Fal Dessai Nilesh B., Naik Gaurav A., Pawar Jyoti D. (2016): Syllabification: An Effective Approach for a TTS System for Konkani, Proceedings of IEEE International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT), Mysuru, PP. 36.

Goibab S. (1040): Konkani Nadshahtra, Vol. 1, pp 392-439.

Kaur G., Singh P. (2013): A Technique to Detect Syllable Boundary in a Wave File, International Journal of Computer Science and Communication Engineering, IJCSCE Special issue on Recent Advances in Engineering & Technology - NCRAET.

Kumar S. M., Prabhu P. E., Reddy M. V. S., Kumar P. M. S. (2014): Text To Speech System for Telugu Language, International Journal of Engineering Research and Applications, Vol. 4, Issue 3 (Version 1).

Linguistics Data Consortium for Indian Languages: http://www.ldcil.org/default.aspx /

Loizou P. C.: Speech Quality Assessment, University of Texas-Dallas, Department of Electrical Engineering, Richardson, TX, USA.

Naser A., Aich D., Amin M., R. (2010): Implementation of Subachan: Bengali Text To Speech Synthesis Software, 6th International Conference on Electrical and Computer Engineering, ICECE, Dhaka, Bangladesh.

Rao N. M., Thomas S., T. Nagarajan, Murthy H. A. (2005): Text-to-Speech Synthesis using syllable-like units, Proceedings of National Conference on Communication (NCC).

Repe M. R. (2010): Natural Prosody Generation in TTS for Marathi Speech Signal, IEEE International Conference on Signal Acquisition and Processing.

Sadeque F. Y., Yasar S., Md. Islam M. (2013), “Bangla Text to Speech Conversion: A Syllabic Unit Selection Approach”, International Conference on Informatics, Electronics and Vision (ICIEV).

Sangeetha J., Jothilakshmi S., Sindhuja S., Ramalingam V. (2013): Text To Speech Synthesis System For Tamil, International Journal of Emerging Technology and Advanced Engineering, Volume 3, Special Issue 1.

Shirbahadurkar S. D., Bormane D. S. (2009): Marathi Language Speech Synthesis Using Concatenative Synthesis Strategy, Machine Vision, ICMV, Second International Conference, Pages: 181 – 185.

Singh P., Lehal G. S.: Text To Speech Synthesis System for Punjabi Language, Proceedings of International Conference on Multidisciplinary Information Sciences and Technologies, Merida, Spain.

Soedirdjo S. D. H.; Zakaria H., Mengko R. (2011): Indonesian text-to-speech using syllable concatenation for PC-based low vision aid, Electrical Engineering and Informatics (ICEEI) International Conference.

Tadkodkar P, Mandgutkar A.: Konkani Parichai, pp 5-21.

Wavesurfer Tool: https://en.wikipedia.org/wiki/WaveSurfer

Zeki M., Khalifa O. O., Naji A. W. (2010): Development of an Arabic text-to-speech system, Computer and Communication Engineering (ICCCE), International Conference, Kuala Lumpur.