A Context-based Numeral Reading Technique for Text to Speech Systems

Soumya Priyadarsini Panda[1], Ajit Kumar Nayak[2]
[1]Department of CSE Silicon Institute of Technology Bhubaneswar, India
[2]Department of CS & IT Siksha 'O' Anusandhan University Bhubaneswar, India

Article Info

Article history:

Received Nov 6, 2017
Revised Jun 22, 2018
Accepted Jun 30, 2018

Keyword:

Concatenative synthesis
Indian languages
Speech synthesis
Text-to-Speech systems
Waveform Concatenation Technique

ABSTRACT

This paper presents a novel technique for context based numeral reading in Indian language text to speech systems. The model uses a set of rules to determine the context of the numeral pronunciation and is being integrated with the waveform concatenation technique to produce speech out of the input text in Indian languages. For this purpose, the three Indian languages Odia, Hindi and Bengali are considered. To analyze the performance of the proposed technique, a set of experiments are performed considering different context of numeral pronunciations and the results are compared with existing syllable-based technique. The results obtained from different experiments shows the effectiveness of the proposed technique in producing intelligible speech out of the entered text utterances compared to the existing technique even with very less storage and execution time.

Corresponding Author:

Soumya Priyadarsini Panda,
Department of CSE,
Silicon Institute of Technology,
Bhubaneswar, Odisha, India.
Email: sppanda.cse@gmail.com

1. INTRODUCTION

The goal of speech synthesis is to develop a machine having an intelligible, natural sounding voice for conveying information to the users in a desired voice, language and accent [1], [2]. Research in the area of speech synthesis is a multi-disciplinary field with applications from acoustic phonetics (speech production and perception) [3] over morphology (pronunciation) [4] and syntax (parts of speech, grammar) [5], to speech signal processing (synthesis) [6]. Recent research in the area of Speech and Language Processing enables machines to speak naturally like humans [7]. A Text-to-Speech (TTS) system in this aspect converts natural language text into its corresponding speech [8]. The intelligible speech synthesis systems have a widespread area of applications in developing human–computer interactive system [9] like, talking computer systems [10], talking toys [11], etc. Speech synthesis, combined with speech recognition, allows for interaction with mobile devices via natural language processing interfaces.[12] Analyzing the input text and converting it into a computer readable form for obtaining the appropriate pronunciation plays an important role in appropriate speech unit production and for its understandability by the listeners [13]. Text analysis is the front end language processor of the TTS system [14], which accepts input text, analyzes it and organizes into manageable list of words [15].

An input text may contain symbols (double quote, comma, report, etc), numbers, abbreviations or special symbols [16]. Text normalization involves transformation of the raw input text into the equivalent of written words [17]. It also involves converting all letters of lowercase or upper case, removing punctuations, accent marks, stop words or too common words (like "Don't" vs. "Do not", "I'm" vs. "I am", "Can't" vs.

"cannot", etc). Sentences are a group of word segments and these segments may be an acronym, a single word or a numeral [18]. While for the abbreviations or acronyms, one time normalized pronunciations may be maintained, the pronunciation of numerals varies depending on the context of its use in the sentence or word [19].

A number may be pronounced differently in different situations and needs to be converted into their appropriate pronounceable forms to produce the desired speech outputs [18]. Table 1 shows some example pronunciation of the English numerals in different situations. In this aspect most of the foreign languages like English are well researched [19], where the pronunciation rules are simpler due to the occurrence of pronunciation repetitions after 20. (e.g.: *Twenty one, twenty two*, …, *Thirty one, Thirty two*,…etc). However, the Indian language TTS techniques sill presents gap for its acceptance by the users due to the unavailability of appropriate pronunciation rules. The probability of repetition of pronunciation is relatively very less in Indian languages at word level (e.g: pronunciations of the numbers in Hindi: 21-"ik-kis", 22-"baa-is", 23-"tei-s", etc.). This increases the complexity of the numeral reading module. Therefore, most of the researchers use simple digit based reading models that stores the recorded units for single digits from 0 to 9 for producing the desired output speech but did not address the context based numeral reading. However, context based numeral reading plays an important role to enhance the understandability of the produced speech. It is always easier to understand the price of some item if it is pronounced based on position based reading like "fifty five thousand five hundred" instead of pronouncing "five five five zero zero". The focus of this paper is to address the context based numeral pronunciation in Indian language scenario.

Table 1. Example Pronunciation of a Number in Different Scenarios

Example	Type	Pronunciation
2015	Date/Quantifier	Two thousand fifteen
2015	Phone number	Two zero one five
0.502	Number	Point five knot two
20.15	Decimal number	Twenty point one five

There are only fewer models documented for speech synthesis in Indian languages [20]-[24], however the context dependent numeral pronunciations has not been well addressed [25]-[28]. The dhvani TTS system for Indian language [25], maintains the pronunciations of numerals up to hundred as the phonetic representation and use the position pronunciations for 'hundred', 'thousand', etc positions attached to the "up to hundred pronunciation" for reading the numerals. However, the context dependent numeral reading aspect is not considered in speech production. A rule-based numeral reading method is presented in [18] for the Odia language.

In this paper, we present a pronunciation rule based approach for the up to hundred pronunciations and incorporate it with the waveform concatenation technique (WCT) [29] to produce output speech for Indian language numerals. Also, the context dependent numeral pronunciation aspects of the numerals are considered to produce natural speech segments to increase the understandability. A set of experiments are performed to evaluate the performance of the proposed model compared to the existing syllable based technique with respect to different contexts of numeral pronunciation. And the results obtained, shows the effectiveness of the proposed technique compared to the existing technique in different contexts.

The remainder of the paper is organized as follows. In the next section, we discussed about the waveform concatenation technique as the proposed numeral reading module is incorporated into the rule based concatenative approach. Section 3 describes the details about the proposed model and the context dependent numeral pronunciation rules. The experimental methodology and result analysis for our technique is given in section 4, showing the effectiveness of this technique in producing intelligible speech. Section 5 concludes the discussion, explaining the findings of our experiments and the future directions of this work, where further work may be undertaken.

2. WAVEFORM CONCATENATION TECHNIQUE (WCT)

As compared to English, most of the Indian languages have approximately twice as many vowels and consonants along with a number of possible conjunct characters formed by combination of two or more characters [28]. Therefore, a large number of speech units are needed to be stored in the speech database while a concatenative speech synthesis technique is used for producing uninterrupted speech. However, WCT [29] uses only 35 basic speech units of the consonant (C) and vowel (V) sounds instead of storing all required speech units in the database, and derive all other units using a rule based waveform concatenation technique. The list of 35 basic speech units are listed in Table 2.

For producing the output speech for the required speech segments, a fraction-based waveform concatenation technique is used. The fraction duarions are determined dynamically from the speech data based on the vowel onset point identification technique [29]. These fractions durations are considered for the waveform concatenation process to obtain the desired speech units. While the rule-based concatenative technique (RCT) [28] uses a static fraction duration for concatenation the use of dynamic fraction durations in WCT [29] enhances the quality of speech being produced. This fraction based concatenation process is considered for the dependent type of unit pairs such as Consonants attached to Matra/Fala/Halant/Consonants and the whole wavedata is used for producing the independent unit pairs like Consonants attached to Consonant/ Vowel, Vowels attacched to Consonants/Vowels. Figure 1 shows the portion based waveform concatenation process to produce the sound "\re" from *"\ra"* and *"\ae"* using the WCT technique.

Table 2. Speech units in Database

Set of speech units in the database						
\a	\o	\cha	\ttha	\tha	\pha	\lla
\aa	\ka	\chha	\dda	\da	\ba	\la
\ee	\kha	\ja	\ddha	\dha	\bha	\sha
\uu	\ga	\jha	\nna	\na	\ma	\ha
\ae	\gha	\tta	\ta	\pa	\ra	\ya

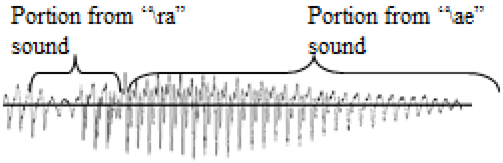


Figure 1. Wave pattern of *"/re"* (C-M) sound after concatenating portions from */ra* and */ae* sound

3. PROPOSED MODEL

In this section, we present a pronunciation rule-based technique for producing speech segments for the Indian language numerals by identifying the phoneme level similarities in the numeral pronunciations in the three considered languages (Odia, Hindi, and Bengali). Figure 2 shows the overview of the proposed numeral reading module and the details of the phases are discussed next. However, first a context identification process is performed to identify the context of the numeral pronunciation as discussed next.

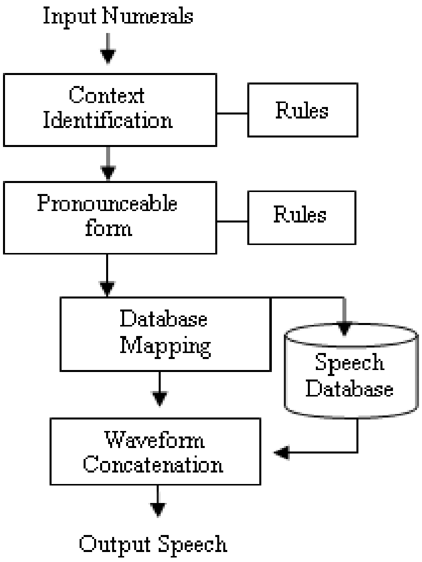


Figure 2. Text-to-speech conversion process

3.1 Context Dependent Numeral Pronunciation

The context dependent numeral pronunciation is an important issue for producing meaningful speech samples for the numerals. The simple digit reading technique may not provide the desired understandability in all situations. For example, while reading a larger quantity or price say 1,54,954 by simply reading the digits as “one-five-four-nine-five-four” makes the listener think to rearrange the numbers to understand the spoken price or quantity; appropriate pronunciation as one-lakh, fifty four-thousand, nine-hundred, fifty-four may make some sense to the listeners. The similar variation of pronunciation also extends to the Indian languages. Table. 3 show some example numerals and their pronunciation in different context in English and Odia language.

Table 3. Pronunciation of a Number in Different Scenarios

Example	Type	English Pronunciation	Odia Pronunciation
୦୨-୦୭-୨୦୧୫ or ୦୨/୦୭/୨୦୧୫	Date	July two two thousand fifteen	“Dui-saat-dui hajaar pandara”
୦୨:୪୦	Time	Two forty	Dui-ta -chaalish
୦୬୭୪-୨୦୧୫	Phone number	Zero six seven four two zero one five	Sun-chha-saat-chaari- dui-sun-aek-paanch
୨୦୧୫	Quantifier	two thousand and fifteen	Dui-hajjar-pandara
୨୦୧୫.୧୫	Number with Decimal points	Two thousand fifteen point one five	Dui-hajjar-pandara-dasmic-pandara

A number in different Indian languages may be pronounced by simply reading the digits while mean for a quantity [21], phone number or credit card number, etc.; the number may be read by the relationship with its positions while meant for a price indicator or year. In case of a fraction value the left part before the decimal point is read based on the relationship between the position of the character and the numbers after the period are read as single digits. While reading a date people always read as “aek-tin-dui-hajaar-sohala” for the date “01-03-2016” in triplet format (dd-mm-yyyy or dd/mm/yyyy). Also, for reading a time interval separated by a colon the format is different for the number before the colon and after the colon. To incorporate all the considered variation of a numeral pronunciation a set of manually coded rules are prepared. The context identification rues are presented below, where n is the number of digits in the number and d_i is the ith digit in the number.

Context dependent pronunciation rules:
Rule 1:
IF n >=10
AND no separation in between
THEN perform digit reading
Rule 2:
IF n >=10
AND d_i separated by “,”
THEN perform position based digit reading
Rule 3:
IF digits separated by “-“or “/” in a triplet format
THEN perform date format digit reading
Rule 4:
IF digits separated by “-“
THEN perform digit reading
Rule 5:
IF digits separated by “:”
THEN perform time format digit reading (digit reading for digits before “:” and position based digits reading for digits after “:”
Rule 6:
IF digits separated by “.”
THEN Perform position based digit reading for digits before”.” and digit reading for digits after “.”
Rule 7:
IF number followed by price indicator
THEN perform position based digit reading
Rule 8:
IF rule not found for the digit format
THEN perform digit reading

3.2. Pronunciation Rules

As the proposed technique for speech synthesis stores only some basic speech units and produce all the sounds from these basic units based on some specified rules, the pronounceable units for numerals are needed to be identified and mapped to the respective character equivalents for the sounds to produce the desired output speech. Also, there is no generalized rule available for the pronunciation of numbers up to 100. However, for numbers greater than 100, a repetition of pronunciation may occur (e.g.: 122 “ek sou-baais”, 123- “ek sou-teis”, etc in Hindi language. Therefore, the numerals after 100 may be formed by concatenating the 100th 1000th,…etc place pronunciations with their respective up to 100 pronunciations. We prepare a set of pronunciation rules for obtaining the up to 100 pronunciations. The pronounceable unit identification process is discussed below.

The numbers from 1-9 and all 10th position pronunciations are needed to be maintained for performing single digit reading. The pronunciations of the numerals from 1-9 and 10th positions are presented in Table 4 and Table 5 respectively for the considered languages. Also, there may be a similarity noticed in the pronunciations of the numerals in the three considered languages.

Table 4. Pronunciation of Numerals up to 10 in the three Considered Languages

Numeral	Odia	Hindi	Bengali
1	Aek	Aek	Aek
2	Dui	Do	Dui
3	Tin	Tin	Tin
4	chaari	Chaar	Chaar
5	Paanch	Paanch	Paach
6	Chha	Chhe	Chhoy
7	saate	Saat	Shaat
8	Aatthe	Aath	Aat
9	na	nau	noy

Table 5. Pronunciation of Numerals for 10th Positions in the three Considered Languages

Numeral	Odia	Hindi	Bengali
0	Sun	Sunya	Shoonno
10	Dasa	Das	Dosh
20	Kodiae	Bish	Kuri/bish
30	Tirish	Tish	Tirish
40	Chaalish	Chaallish	chaallish
50	pachaash	pachaash	ponchaash
60	saathiae	saatth	Shaat
70	saathiae	sattar	shottor
80	asi	ashi	ashi
90	nabe	nabe	nobboi
100	sahe	sau	Sho
1000	hajaare	hajar	hajaar
100000	lakhya	laakh	laksh
10000000	koti	karod	koti

As in Indian languages, the probability of repetition of pronunciation is relatively very less at word level, we try to derive the pronunciation similarities at phoneme level for the up to 100 pronunciations. For example, when the numeral 2 is present at unit or 10th place it has one type of pronunciation at beginning or end. The pronunciation similarities in the three considered languages for 2 at unit and 10th place are presented in Table 6 and Table 7 respectively. Considering such similarities in pronunciations a set of similarity rules are prepared for the pronunciations for numerals from 11-99.

Table 6. Example SCRIPTs and Pronunciation Repetitions for “2” at tenth Place

Numeral	Odia	Hindi	Bengali
21	Eko-is	Ik-kis	Aek
22	Baa-ish	Baa-ish	Baa-ish
23	Te-ish	Te-ish	Te-ish
…	…	…	…
29	Ana-tir-ish	Un-t-ish	unotirish

Table 7. Example Scripts and Pronunciation Repetitions for “2” at unit Place

Numeral	Odia	Hindi	Bengali
12	Baa-ra	Baa-ra	baro
22	Baa-ish	Baa-ish	baaish
32	Ba-tish	Bat-ish	bottrish
…	…	…	…
92	Baya-nabe	bayanabe	Bira-nobboi

The pronunciations of numerals with starting and ending similarities in the three considered languages are presented in Table 8, Table 9 and Table 10 respectively for Odia, Hindi, and Bengali language and for each of the language and similarity three states are maintained at phoneme level as shown in Figure 3 and based on proper match the respective pronounceable units are extracted from the speech database to produce the output speech.

Table 8. Pronunciation of Odia Numerals with Starting (Column Wise) and Ending (Row Wise) Similarity

Pronunciation	aek	baa	te	chau	pan	chhau	sat	ath	ana
Ra	aeg-aa-ra	Baa-ra	Te-ra	Chau-da	pan-da-ra	so-ha-la	sat-a-ra	Ath-a-ra	Une-ish
Is	aek-oi-s	Baa-is	te-tis	Chau-bis	pan- chis	chha -bis	sat-e-is	ath- e-is	Ana-tiris
Ris	aek-ti-ris	ba- tis	te- tis	chau- ti-ris	pain- tiris	chha-tis	sain-tiris -	ath-a tiris	Ana-chalis
Lis	aek-chalis	ba-ya-lis	te-ya--lis	chau-ra-lis	pain-chalis	chha- ya-lis	sat-chalis	ath-a-chalish	Ana-chas
Ban	aek-ban	ba- ban	te- pan	Chau--ban	pan-chaaban -	Chha-pan	sat- aa-ban	ath- aa-ban	Ana-sathi
Sathi	aek-a-sathi	ba- sathi	te-sathi	chau-sathi	pan- sathi	chha-sathi	sat-sathi	ath-a - sathi	Ana-stori
Stari	aek-a-stari	baa-stari	te- stari	chau- stari	pan-cha - stari	chha-stari	sat-a-stari	ath-a-stari	Ana- asi
Asi	aek-a-asi	ba-ya-asi	te-ya-asi	chau- raa-asi	pan- chaa-asi	chha-yaa-asi	sat-aa-asi	ath-aa-asi	Anaa-nabe
Nabe	aek-aa-nabe	ba-ya-nabe	te-ya-nabe	chau-ra-nabe	pan-chaa-nabe	chha-yaa-nabe	sat-aa-nabe	ath-aa-nabe	ane-sat

Table 9. Pronunciation of Hindi Numerals with Starting (Column Wise) and Ending (Row Wise) Similarity

Pronunciation	ik	baa	te	chau	pan	chha	sat	ath	un
Raa	Gya-raa	baa-raa	te-raa	Chau- daa	pan-d-ra	So-la	sat-raa	ath-aa-raa	un-ish
Is	ik-is-is	Baa- is	te-is	Chau-bis	pa-ch-is	Chha-bis	sat-aa-is	ath-aa-is	Un-ti-sh
Tis	ik-a-tis	ba-tis	te-tis	chau-tis	pain-tis	Chha-tis	sain- tis	ath-tis	un-chalis
Lis	ik-cha- -lis	ba-ya-lis	te-ya-lis	chau-ra-lis	pain-ta-lis	Chha- ya-lis	sat-cha-lis	ath-cha--lish	Un-chas
Ban	ik-ya-ban	ba-ban	te- pan	Chau-ban	pa-ch- pan	Chha-pan	sat-aa-ban	ath-aa-ban	un- sath
Sath	ik-sath	ba- sath	te- sath	chau- sath	pain-sath	chha- sath	sat- sath -	ath-a-sath	Un- atar
Tar	ik-a-tar	baa- tar	te- tar	chau- tar	pa-cha- tar	Chha-tar	sat-a-tar	ath-a-tar	una-asi
Asi	ik-ya-asi	ba-ya -asi	te-ya-asi	chau- raa-asi	pan- chaa-asi	Chha-yaa-asi	sat-aa-asi	ath-aa-asi -	un-ya-nabe
Nbe	ik-ya-nbe	ba-ya - nbe	te-ya-nbe	chau-ra-nbe	pan-chaa-nbe	chha-yaa - nbe	sat-aa-nbe	ath-aa-nbe	ni-nya--nbe

To derive all the pronunciations for the numerals, we have prepared different groups considering the above discussed similarities. We have classified the pronounceable units to be into three states of groups as: Begin state (B), Middle state (M) and End state (E). Depending on the position of the number i.e. unit or 10th, the states are determined and the pronunciation is derived. For example for obtaining the pronunciation of a number N having length L, as $\{n_1, n_2, \ldots n_L\}$, There exist 3 states representatives of the pronunciations, {B, M, E} for the unit and 10th positions, n_L and n_{L-1} respectively and the units from n_1 to n_{L-1} may be derived using the common pronunciation rules by concatenating 100th, 1000th, etc position’s pronunciation with the up to 100 pronunciation. For example, in producing the pronunciation of the numeral 11 as “*ek-ga-ra*” in

Odia language the units involved in the pronunciation are “\ek” from “B set”, “\ga” from “M set” and “\ra” from “E set” as shown in Figure 3. i.e B(L)-M(L-1)-E(L-1). However the above repetitive pronunciation is not same for numbers having a 9 or 0 at the unit place. To overcome this we separate the numbers of these category from the groups and produce their pronunciation by maintaining special cases of pronunciation as {9: *“na”,* 19: *“une-is”,* 29: “*ana-tiris*”, etc}, {20: *“ko-die”*, 30: *“tiris”*, etc} However, for some units, the pronunciation rules does not include the middle state, for example for the numeral two at the unit place, the mapping may be “\ba” from the B state and the next one is from the E state as “\ra” to form the pronunciation “ba-ra” for the numeral “12”.

Table 10. Pronunciation of Bengali Numerals with Starting (Column Wise) and Ending (Row Wise) Similarity

Pronunciation	aek	baa	te	cho	pon	chho	shat	ath	una
ra	Aeg-aa-ro	Baa-ro	te -ro	cho- ddo	pon- e-ro	so- ha-la	shat- a-ra	ath- a-ra	Une-ish
ish	aek-ush	baa-ish	te-ish	cho-bish	po-ch-ish	chho-bish	shat- aa-sh	ath-aa-sh	Uno-tirish
rish	aek-ti -rish	bo-t- rish	te-ti-rish	chou-ti-rish	poy-ti-rish	chho-ti-rish	shai- ti-rish -	att-i rish	Uno-cholish
lish	aek-cho -lish	bi-ya-lish	te-ta-lish	chu-ya-lish	poy-ta-lish	chhe-chho-lish	shat-cho-lish -	att-cho lish	uno-pon-chaash
nno	aekaa -nno	Baha-nno	ti-panno	chu-ya-nno	pon-cha-nno	chha-panno	shat- aa-nno	att-aa-nno	uno-shat
shotti	aek-shotti	ba- shotti	te-shotti	chou-shotti	poy-shotti	chhe-shotti	shat-shotti	att- shotti	Uno-shottor
ttor	aek-a-ttor	baha- ttor	ti-ya ttor	chu-ya-ttor	po-cha-ttor	chhi-ya-ttor	shat-aa-ttor	att-aa-ttor	Uno-ashi
ashi	aek-aa -ashi	bi-r-ashi	tir- ashi	chu-ra-ashi	po-cha-ashi	chhi- ya-ashi	shat-a-ashi	ata- ashi	uno-no-bboi
nobboi	aek-aa-nobboi	bi-ra-nobboi	tir-a-nobboi	chu-ra-nobboi	po-chaa-nobboi	chhi-yaa-nobboi	shat-aa-nobboi	ataa-nobboi	nira-nobboi

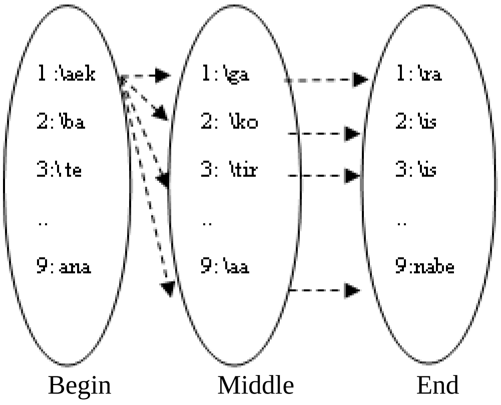


Figure 3. Possible states of a numeral in Odia language

The upto 100 pronunciation may fail for certain numerals, e.g: consider the numeral 14 pronounced as “chau-da”. This does not follow the pronunciation similarities. An obvious (brute force) workaround is to have a small dictionary of such dis-similar units, and check whether a given number matches any of them at the beginning of text analysis phase. If so, break it up into the corresponding pronounceable units separately and parse them to the next phase separately. This works satisfactorily, and we've implemented this with a few numerals (14-“chau-da”, 16- “so-ha-la”, 35-“pain-tir-is”, 53- “te-pan”, 56- “chha-pan”, etc).

3.3. Speech Database Mapping And Waveform Concatenation

As the model uses the WCT technique to produce the desired output speech, the respective base sound units in the speech database are needed to be obtained for performing waveform concatenation to produce the output speech. The speech database mapping phase identifies the respective speech database units to perform rule based waveform concatenation. The WCT technique is then used to produce the desired output speech for the Odia numeral. Figure 4 shows the portion concatenation process for the numeral “1” pronounced as “ae-ka” in Odia language from the two speech database units“\ae” and “\ka”.

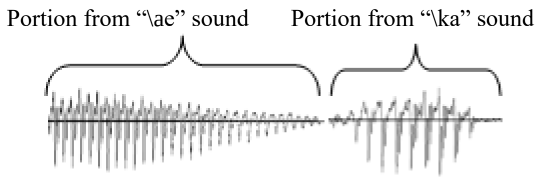
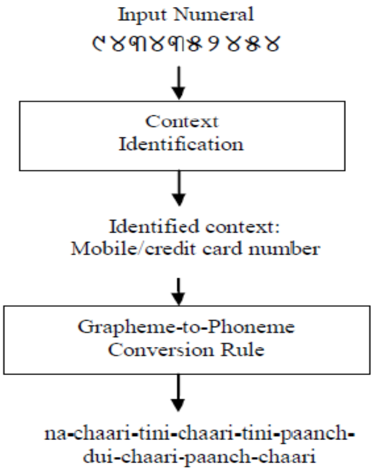


Figure 4. Wave pattern of numeral “1” (one) in Odia (‘୧’) pronounced as “ae-ka”

3.4. ILLUSTRATION

The context identification process for an Odia language numeral is presented in Figure 5 and the speech unit identification/mapping step involved for producing the numeral pronunciation from the base 35 speech units is presented in Figure 6. In producing the output speech for the numerals, the discussed up to 100 pronunciation rule is used to find the equivalent character units involved. The same portion concatenation method is used to produce the final output speech.


Input Numeral
୯୪୩୪୩୫୨ ୪୫୪
Context Identification
Identified context: Mobile/credit card number
Grapheme-to-Phoneme Conversion Rule
na-chaari-tini-chaari-tini-paanch-dui-chaari-paanch-chaari

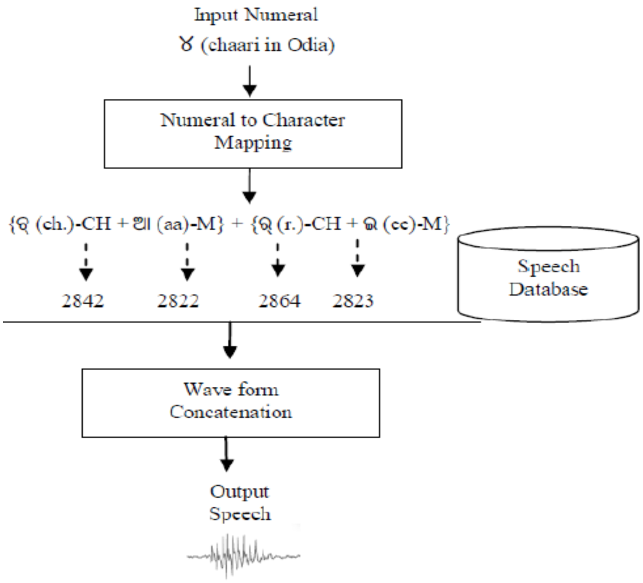


Figure 5. Numeral context identification for input numeral 9434352454 in Odia language

Figure 6. Numeral sound production for input numeral 4 in Odia language

4. RESULT ANALYSIS

The proposed numeral reading technique and the WCT technique is implemented in C/C++ and is being tested for producing different types of numerals in different context in the considered Indian languages. To analyze the quality of the produced speech, the Mean Opinion Score (MOS) test [30] is considered along with the storage and execution time with respect to the existing syllable based text to speech technique [19]. The details of the results obtained are discussed below.

4.1. Storage Requirement

While the syllable based techniques requires around 800 speech units of syllable units requiring a memory of around 1-2 MB in compressed format, the WCT technique that produces the speech segments from the basic 35 speech units requiring a memory of around 235 KB only without further compressions. No other units are required to be added to the database for producing the numeral pronunciation in different context. Assuming the total storage required by the syllable-based technique to be 100%, the proposed technique achieves 81% reduction in total storage requirement and 91% reduction in number of units in the speech database compared to the existing technique.

4.2. Execution Time

To analyze the performance of the proposed technique in terms of execution time compared to the syllable based technique, different text files are prepared containing numerals in different context of its use. By varying the number of numerals in each file from 10 to 100, the execution time (in ms) is measured by both the techniques. Figure 10, Figure 11 and Figure 12 shows the average execution time for both the techniques. The results all the experiments performed shows the exponential increase of execution time, due to the increase in number of decompression to the .gsm files in the syllable based technique, while the proposed approach shows relatively very low growth rate in all the scenarios tested.

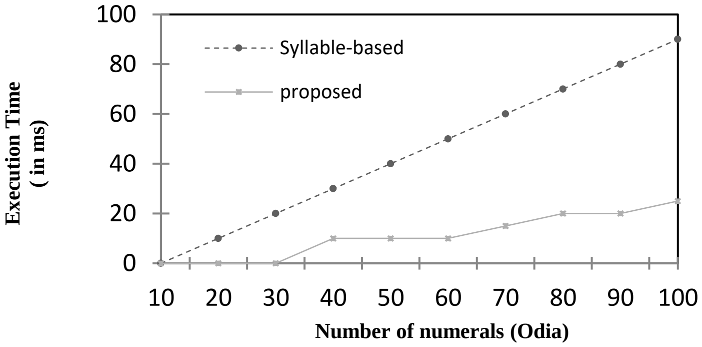


Figure 10. Execution time for syllable-based and proposed technique with respect to increasing number of words in Odia language

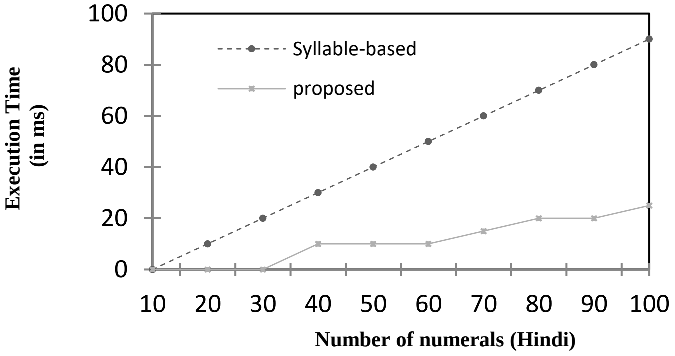


Figure 11. Execution time for syllable-based and proposed technique with respect to increasing number of words in Hindi language

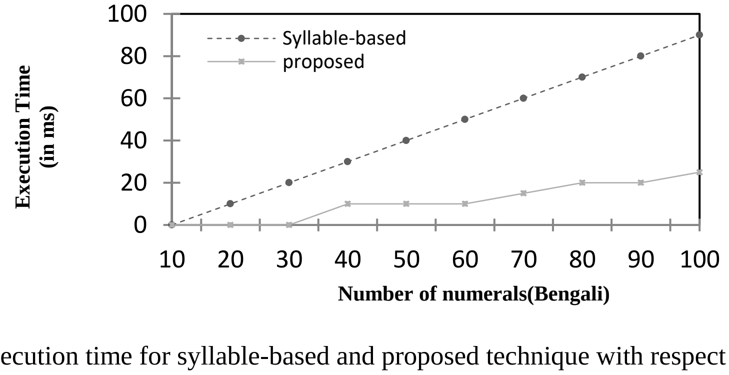


Figure 12. Execution time for syllable-based and proposed technique with respect to increasing number of words in Bengali language

4.1. Subjective Measure For Speech Quality

For performing the MOS tests, a set of random numerals, N_1, N_2….N_8 are selected representing different category of pronunciations for the specified rules. The output speech is generated by the proposed technique as well as by the syllable-based numeral reading technique. A group of 15 native speakers are selected from each language to perform the listeners test and are asked to give their feedback on the basis of ease of understandability on output speech produced by the two techniques in a 5 point scale (1-very low, 2-low, 3-average, 4-high, 5: very high). All the tests were performed with a headphone set. Figure.7, Fig. 8 and Fig. 9 shows the average MOS test results by all listeners for different numerals respectively for the three considered languages. The results of all the experiments performed show the effectiveness of the proposed technique in producing comparable results with the existing technique even with a very small database.

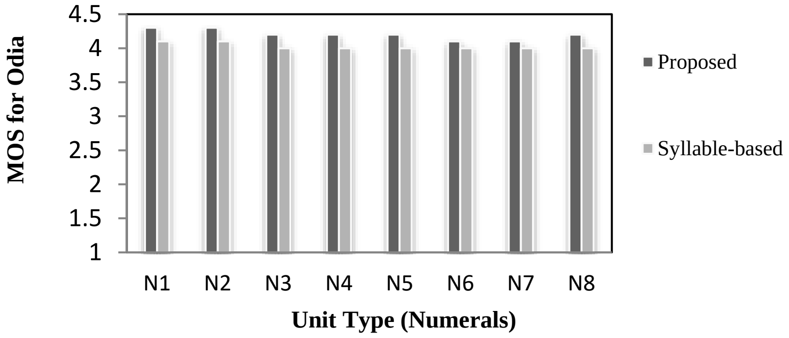


Figure 7. Average MOS for Odia language

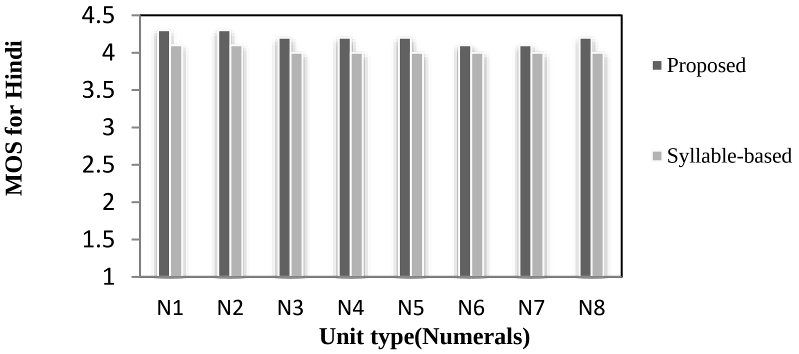


Figure 8. Average MOS for Hindi language

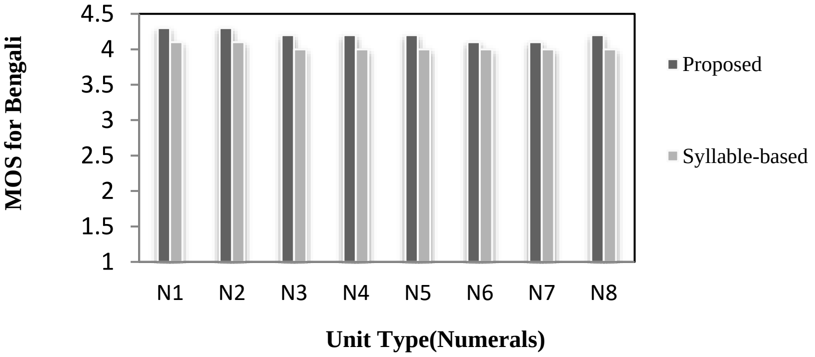


Figure 9. Average MOS for Bengali language

5. CONCLUSIONS

In this paper, a context based numeral reading technique is presented for Indian language text to speech systems. The proposed pronunciation rule based model is incorporated with the WCT technique to produce the desired output speech. To evaluate the performance of the proposed technique, a set of experiments were performed to show the effectiveness of the technique compared to the existing syllable-based technique. The subjective measure analysis shows the effectiveness of the proposed technique in producing intelligible speech compared to the existing technique, even with a very small speech database of 35 basic units only. The average execution time required by the proposed technique is also very less compared to the exiting technique. However, the model provides the pronunciation rules for three Indian languages only while the same level of similarity may be observed in other Indian languages. Therefore, the model may further be enhanced to work for other Indian languages. Also, some smoothening techniques may be applied at the concatenation points to further enhance the quality of speech being produce to make it more natural sounding.

REFERENCES

[1] J. Feng, B. Ramabhadran, John H. L. Hansen, and J. D. Williams, "Trends in Speech and Language Processing," *IEEE Signal Processing Magazine,* Jan 2012.

[2] S. Ahmed, and K. Senthil, "Interaction with ATM for Blind," I*ndonesian Journal of Electrical Engineering and Computer Science,* vol. 9, no. 3, 2018.

[3] T. S. Gunawan, M. F. Alghifari, M. A. Morshid, M. Kartiwi, "A Review on Emotion Recognition Algorithms using Speech Analysis," *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, vol. 6, no. 1, pp. 12-20, 2018

[4] S. Singh, "Forensic and Automatic Speaker Recognition System," *International Journal of Electrical and Computer Engineering (IJECE),* vol. *8,* no. 5, 2018.

[5] A. Sulong, T. S. Gunawan, O. O. Khalifa, M. Kartiwi, H. Dao, "Single Channel Speech Enhancement using Wiener Filter and Compressive Sensing," *International Journal of Electrical and Computer Engineering,* vol. 7, no. 4, 2017.

[6] P. Doungpaisan, and A. Mingkhwan, "Query by Example of Speaker Audio Signals using Power Spectrum and MFCCs," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. *7, no.* 6, pp. 3369-3384, 2017

[7] M. F. Alghifari, T. S. Gunawan, and M. Kartiwi, "Speech Emotion Recognition Using Deep Feedforward Neural Network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. *10*, no. 2, pp. 554-561, 2018

[8] [8] F. Alías, X. Sevillano, J. C. Socoró, and X. Gonzalvo, "Towards High-Quality Next-Generation Text-to-Speech Synthesis: A Multidomain Approach by Automatic Domain Classification," IEEE Transactions on *Audio, Speech and Language Processing,* vol. 16, no. 7, 2008.

[9] A. Sulong, S. G. Teddy, O. O. Khalifa, M. Kartiwi, H. Dao , "Single Channel Speech Enhancement using Wiener Filter and Compressive Sensing," International Journal of Electrical and Computer Engineering, vol. 7, no. 4, pp. 1941-1951, 2017.

[10] P. Doungpaisan, and A. Mingkhwan, "Query by Example of Speaker Audio Signals using Power Spectrum and MFCCs," International Journal of Electrical and Computer Engineering (IJECE), vol. 7, no. 6, 2017.

[11] S. Osmanaj, A. Shala, and B. Prevalla, (2017). "The Effect of Bandwidth on Speech Intelligibility in Albanian Language by Using Multimedia Applications like Skype and Viber," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 5, pp. 2514-2519, 2017.

[12] N. Nukaga, R. Kamoshida, K. Nagamatsu, and Y. Kitahara, "Scalable Implementation of Unit Selection Based Text-to-Speech System for Embedded Solutions," *ICASSP*, IEEE , 2006

[13] A. K. Raj, T. Sarkar, S. C. Pammi, S. Yuvaraj, M. Bansal, K. Prahallad, A. W. Black, "Text Processing for Text-to-Speech Systems in Indian Languages," *6th ISCA Workshop on Speech Synthesis, 2007*.

[14] M. Rojc, and Z. Kačič, "Time and space-efficient architecture for a corpus-based text-to-speech synthesis system," Speech Communication, vol. 49, no. 3, pp. 230-249, 2007.

[15] S. Tiomkin, D. Malah, S. Shechtman, Z. Kons, "A Hybrid Text-to-Speech System That Combines Concatenative and Statistical Synthesis Units," *Audio, Speech and Language Processing,* IEEE, vol. 19, no. 5, 2011

[16] J. York, P. C. Pendharkar, "Human–computer interaction issues for mobile computing in a variable work context," International Journal of Human-Computer Studies, vol. 60, no. 5, pp. 771-797, 2004.

[17] S. P. Panda, A. K. Nayak, and S. Patnaik, "Text-to-speech synthesis with an Indian language perspective," *International Journal of Grid and Utility Computing*, vol. *6*, no. 3-4, pp. 170-178, 2015

[18] S. P. Panda, and A. K. Nayak, "A Pronunciation Rule-Based Speech Synthesis Technique for Odia Numerals," Computational Intelligence in Data Mining, pp. 483-491, 2016

[19] N. P. Narendra, K. S. Rao, K. Ghosh, R. R. Vempada, and S. Maity, "Development of syllable-based text to speech synthesis system in Bengali," International Journal of Speech Technology, vol.14, no.3, pp.167-181, 2011.

[20] S. Talesara, H. A. Patil, T. Patel, H. Sailor, N. A. Shah, "Novel Gaussian Filter-based Automatic Labeling of Speech Data for TTS System in Gujarati Language," ICALP proceedings, pp.139-142, 2013.

[21] M. Toman, M. Pucher, S. Moosmüller and D. Schabus, "Unsupervised and phonologically controlled interpolation of Austrian German language varieties for speech synthesis," Speech Communication, vol. 72, pp. 176-193, 2015.

[22] S. Thomas, M. N. Rao, H. Murthy, and C. S. Ramalingam, "Natural sounding TTS based on syllable-like units," 14th European Signal Processing Conference, IEEE, pp. 1-5, 2006.

[23] J. Rama, A. G. Ramakrishnan, R. Muralishankar, R. Prathibha, "A complete text-to-speech synthesis system in Tamil," WSS'proceedings, pp.191-194, 2002.

[24] V. R. Reddy, and K. S. Rao, "Two-stage intonation modeling using feed forward neural networks for syllable based text-to-speech synthesis," Computer Speech & Language, vol. 27, no. 5, pp. 1105-1126, 2013.

[25] http://dhvani.sourceforge.net/ 10th August 2017

[26] S. P. Panda, and A. k. Nayak, "An efficient model for text-to-speech synthesis in Indian language," International Journal of Speech Technology, vol. 18, no. 3, pp. 305-315, 2015.

[27] H. M. Torres, J. A. Gurlekian, "Acoustic speech unit segmentation for concatenative synthesis," Computer Speech & Language, vol. 22, no. 2, pp. 196-206, 2008.

[28] S. P. Panda, and A. k. Nayak, "A Rule-Based Concatenative Approach to Speech Synthesis in Indian Language Text-to-Speech Systems," in proc: ICCD, pp.523-531, 2014

[29] S. P. Panda, and A. K. Nayak, "A waveform concatenation technique for text-to-speech synthesis," International Journal of Speech Technology, vol. 20, no. 4, pp. 959-976, 2017.

[30] M. Viswanathan, "Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (MOS) scale," Computer Speech and Language, vol. 19, pp. 55–83, 2005.