ISSN 2319-2720

Volume 1, No.2, September – October 2012

International Journal of Computing,Communications and Networking

Available Online at http://warse.org/pdfs/ijccn01122012.pdf

A Rule Based Algorithm for Automatic Syllabification of a Word of Bodo Language

Chandan Sarma[1], Jyotismita Talukdar[2,]Prof.P.H Talukdar[3]

[1]Gauhati University,India,chandan.rrc@gmail.com

[2]Asian Institute of Technology, Gauhati University,India,[3]phtassam@gmail.com

ABSTRACT

The process of syllabification performs the task of Identifying syllables in a word. The correct Syllabification rules and algorithms are mainly used in text-to-speech system to improve naturalness of the synthesized speech. This paper presents a study of Bodo syllable structure and linguistic rules for syllabification as well. An algorithm has been developed for automatic syllabification of Bodo language. For evaluation, the algorithm was tested on 5000 words obtained from a corpus and compared with the same words manually syllabified. The algorithm performs with 97.05% accuracy.

Key words : Text-to speech, Phoneme, Syllable.

1. INTRODUCTION

Syllable is a unit which is intermediate between phoneme and word, larger than phoneme and smaller than word[1]. So, many theories are available in phonetics and phonology to define a syllable [2]. In phonetics, the syllables are defined based upon the articulation [3]. But in phonology, the syllables are termed as the different sequences of the phonemes.

Syllabification is the process of dividing a word into its constituent syllables [4]. Syllabification in a TTS system is essential for two reasons [5]. First, it helps the implementation of certain letter-to-phoneme rules. Second, syllabification is essential in enhancing the quality of synthetic speech since detecting the syllable will help in modeling duration and improve the synthesized speech intonation. The quality of a TTS system is measured by the extent of naturalness of the synthetic speech produced by the system [6]. Syllabification helps to improve the naturalness of the synthetic speech [7].

Text-to-speech (TTS) systems are considered not just very innovative, but also a very mature technology in the speech area [3]. It permits automatic synthesis of speech from text. To make these kinds of systems robust, efficient and reliable, for the texts to be synthesized. The pre-processing module (also known as front-end) of a TTS system is composed by three stages: text analysis, phonetic analysis and prosodic generation. The syllabification of a word is a task for which the text analysis stage is responsible.

In this work, an algorithm to syllabify Bodo word into syllables is proposed. The algorithm was tested by using a text corpus containing representative words for each grammatical rule, and its performance was then measured in terms of the percentage of correctly syllabified words.

2. PHONOLOGICAL STRUCTURE OF BODO LANGUAGE

Bodo is a language that belongs to the branch of Barish section under Baric division of the Tibeto-Burman languages and spoken by the Bodo people of north-eastern India and Nepal. The language is one of the official languages of the Indian state of Assam, and is one of the 22 scheduled languages that is given a special constitutional status in India. The number of Bodo people to speak Bodo language is estimated at 15 lakhs inhabiting throughout the plains of Assam, particularly throughout the thickly Bodo populated areas of the northern parts of the state extending from Dhubri District in the west to Lakhimpur District in the east. It is used as language media or lingua franca also in the northern parts particularly in districts of Kokrajhar, Dhubri, Bongaigaon, and some other sub-divisional and pocket areas like Bijni, Rangia, Udalguri, Barama, Dudhnai, Boko etc.

The inventory of Bodo phonemes consists of the number of (a) segmental phonemes being consonants and vowels and (b) suprasegmental phonemes being tone, juncture and contour, co-occurring with them as extra sound features used in the language [9]. Bodo language contains 22 segmental phonemes; six pure vowels or monophthongs and sixteen consonants including two semi vowels.

The following Table 1 and Table 2 displays the Bodo vowels and consonants.

Table 1: Bodo Vowels

	Front	Central	Back
Close	**I**		**u w**
Mid	**E**		**Ɔ**
Open		**a**	

Table 2: Bodo Consonalts

Manner of articulation		Bilabial		Alveolar		Alveolo-Palatal		Velar		Glottal
		Vl	**Vd**	**Vl**	**Vd**	**Vl**	**Vd**	**Vl**	**Vd**	**Vd**
STOP			b		d				g	
		p^h		t^h				k^h		
Nasal			m		n				ŋ	
Fricative						s	z			h
Trill					r					
Lateral					l					
Semi-vowel			w				j			

3. SYLLABFICATION

In phonology syllables are termed as different sequences of phoneme. Most linguists view syllables as an important unit of prosody because many phonological rules and constraints apply within syllables or at syllable boundaries [10]. Apart from their purely linguistic significance, syllables play an important role in speech synthesis and recognition [11]. The pronunciation of a given phoneme tends to vary depending on its location within a syllable. While actual implementations vary, text-to-speech (TTS) systems must have, at minimum, three components [12] : a letter-to-phoneme (L2P) module, a prosody module, and a synthesis module. Syllabification can play a role in all three.

3.1 Methodology

The methodology adopted in this study was to first examine the Bodo Syllable structures from the Linguistics literature to gather the opinions of scholars from the various linguistic traditions. It was expected that this study would reveal the main issues related to the syllabification of Bodo language and how these issues are addressed by scholars in the literature. We had closely examined the syllabification rules also. Based on the rules an algorithm was developed to syllabify Bodo words automatically and the results were evaluated.

3.2 Syllable Structure in Bodo

The Bodo language is highly monosyllabic. Basically, Bodo words may be either monosyllabic or polysyllabic, co-occurring with either rising tones or falling tones.

The syllables are described on the basis of the sequences of phonemes in segments of the vowels(V) and the consonants(C) and also of the clusters in consonants. The Bodo syllable structures may be divided into the following types based on the distribution of the segmental phonemes and consonantal clusters-

1. V
2. VC
3. CCV
4. CCVC
5. CVCVCC
6. CVCCCVC
7. CVCCVCCVCCV

3.3 Syllabification Procedure for Bodo Language

In Bodo language every syllable has one vowel sound. So, number of vowel sounds in a word equals to the number of syllable. A monosyllabic word (e.g /ai/) need not to be syllabified and consonants blends and digraphs (/kh/, /ph/) also need not to be syllabified. Following are the basic rules of syllabification in Bodo.

1) If there is a single vowel in the word then mark the syllable boundary at the end of the word.(Rule #1).

2) If vowels in a word have different sounds then mark the syllable boundary after the first vowel.ie if VV then V/V. (Rule #2).

3) If a word has consonant-vowel structure like VCV means one consonant is there between two vowels then mark the syllable boundary after the first vowel.ie if VCV then V/CV. (Rule #3).

4) If a word has consonant-vowel structure like VCCV means two consonants exist between two vowels then mark the syllable boundary between the consonants.ie if VCCV then VC/CV.(Rule #4).

5) If three consonants comes between two vowels in a word like VCCCV then mark the syllable boundary after the first consonants.ie if VCCCV then VC/CCV.(Rule #5)

6) If a word has consonant-vowel structure like CVVC means two vowels come between two consonants then mark the syllable boundary between the vowels forming two syllables.ie if CVVC then CV/VC.(Rule #6)

7) If a word has consonant-vowel structure like CVVCV means if two vowels come between two consonants and the last consonant followed by another vowel mark the syllable boundary after two consecutive vowels.ie if CVVCV then CVV/CV.(Rule #7).

8) If a word has consonant-vowel structure like VCVC or VCVV means VCV followed by either C or V then mark syllable boundary after the first vowel.ie if VCVC or VCVV then V/CVC or V/CVV.(Rule #8).
9) If a word has consonant-vowel structure like CVCVV or CVCVC means CVC followed by either VV or VC then mark the syllable boundary between first vowel and the consonant followed by it.ie if CVCVV or CVCVC then CV/CVV or CV/CVC. (Rule# 9).

10) If a word has consonant-vowel structure like CVCCV means if CVC is followed by CV then mark the syllable boundary between two consecutive consonants.ie if CVCCV then CVC/CV.(Rule #10) .

3.4 Syllabification Action

All the rules were implemented in the algorithm to be inside a loop that executes as many iterations as possible to have all the syllables of a given word separated, always analyzing such word from left to right. On each iteration, the algorithm attempts to find a new syllable in the portion of the word that has not yet been analyzed by trying to match one of the rules with this entire portion or just part of it, always obeying the hierarchical sequence for the verification of these rules defined above. When a rule is matched, depending on its definition, the algorithm can take an action to extract a new syllable from that part of the word that is currently under analysis.

4. SYLLABIFICATION ALGORITHM

In this section, the Bodo syllabification rules identified in the section 3.3 are presented in the form of a formal *algorithm*. The function ***syllabify()*** accepts an array of phonemes generated, along with a variable called ***current_index*** which is used to determine the position of the given array currently being processed by the algorithm.

Initially the ***current_index*** variable will be initialized to 0. The ***syllabify()*** functionis called recursively until all phonemes in the array are processed. The function ***mark_syllable_boundary(position)*** will mark the syllable boundaries of an accepted array of phonemes. The other functions used within the ***syllabify()*** function are described below.

- ***total_vowels(phonemes)****:* accepts an array of phonemes and returns the number of vowels contained in that array.

- *is_vowel(phoneme):* accepts a phoneme and returns true if the given phoneme is a vowel.

- *count_no_of_consonants_upto_next_vowel(phonemes, position):* accepts an array of phonemes and a starting position; and returns the count of consonants from the starting position of the given array until the next vowel is found.

The complete listing of the algorithm is as follows:

```
function  syllabify (phonemes, current_index)

if total_vowels(phonemes) is 1 then
   mark_syllable_boundary(at_the_end_of_phonemes)
else
   if  is_vowel(phonemes[current_index]) is true then
         total_consonants= count_no_of_consonants_upto_next_vowel (phonemes,current_index)
       if   total_consonants is 0 then
          if  is_vowel(phonemes[current_index+1]) is true then
              if is_vowel(phonemes[current_index+3) is true then
                 mark_syllable_boundary(current_index+1)
                 syllabify(phonemes,current_index+2)
              end if
              mark_syllable_boundary(current_index)
              syllabify(phonemes, current_index+1)
          end if
       else
          if total_consonants is 1 then
             mark_syllable_boundary(current_index)
             syllabify(phonemes, current_index + 2)
          end if
          if no_of_consonants are 2 then
              mark_syllable_boundary(current_index+1)
              syllabify(phonemes, current_index+3)
          end if
          if  total_consonants are 3 then
              mark_syllable_boundary(current_index+1)
              syllabify(phonemes,current_index+4)
         end if
       end if
   else
       syllabify(phonemes,current_index+1)
   end if
```

5. RESULTS AND DISCUSSION

The above algorithm was tested on 5000 distinct words extracted from a Bodo corpus and then compared with correctly hand syllabified words to measure accuracy.
Text obtained from the category News Paper, Feature Articles etc was chosen for testing the algorithm due to the heterogeneous nature of these texts and hence the perceived better representation of the language. A list of distinct words was first extracted, and the 5000 most frequently occurring words chosen for testing the algorithm.

The 5000 words yielded some **18,755 syllables** .The algorithm achieves an overall accuracy of **97.05%** when compared with the same words manually syllabified by an expert.

An error analysis revealed that foreign words directly encoded in Bodo produces error.

6. CONCLUSION

Syllabification is an important component of many speech and language processing systems, and this algorithm is expected to be a significant contribution to the field, and especially to researchers working on various aspects of the Sinhala language.

REFERENCES

1. ChandanSarma, U.Sharma,C.K. Nath, S.Kalita, and P.H.Talukdar. **Selection of Units and Development of Speech Database for Natural Sounding Bodo TTS System**,CISP Guwahati, March 2012.

2. Parminder Singh and Gurpreet Singh Lehal. **Syllables Selection for the Development of Speech Database for Punjabi TTS System**, IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 6, November 2010.

3. R.A. Krakow. **Physiological organization of syllables: A Review**, Journal of Phonetics, Vol. 27, 1999, pp. 23-54.

4. Susan Bartlett, Grzegorz Kondrak and Colin Cherry. **On the Syllabification of Phonemes**, Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL, pp.308–316, Boulder, Colorado, June 2009. c 2009 Association for Computational Linguistics.

5. Y. A. El-Imam. **Phonetization of Arabic: Rules and Algorithms.** Computer Speech & Language, Vol. 18, pp.339–373, October 2004.

6. A.W. Black and K.A. Lenzo. **Building synthetic voice**, http://festvox.org/bsv/, 2003

7. R. Dale et al. (Eds.), **A Rule Based Syllabification Algorithm for Sinhala**, IJCNLP 2005, LNAI 3651, pp.438 – 449, 2005.© Springer-Verlag Berlin Heidelberg 2005.

8. Couto I, Neto N, Tadaiesky, V, Klautau A and Maia,R.2010. **An open source HMM-based text-to-speech system for Brazilian Portuguese**. Proc. 7th International Telecommunications Symposium Manaus.

9. Madhu Ram Baro, **Structure of Boro language**,2008

10. Juliette Blevins. **The syllable in phonological theory**,1995

11. George Kiraz and Bernd M¨obius. **Multilingual syllabification using weighted finite-state transducers,** In Proceedings of the 3rd Workshop on Speech Synthesis,1998.

12. Robert Damper. 2001. **Learning about speech from data: Beyond NETtalk.** In Data-Driven Techniques in Speech Synthesis, pp. 1–25. Kluwer Academic Publishers.