Interspeech 2018
2-6 September 2018, Hyderabad

Resyllabification in Indian Languages and its Implications in Text-to-speech Systems

Mahesh M, Jeena J Prakash, and Hema A Murthy

Department of Computer Science and Engineering
Indian Institute of Technology Madras
hema@iitm.ac.in

Abstract

Resyllabification is a phonological process in continuous speech in which the coda of a syllable is converted into the onset of the following syllable, either in the same word or in the subsequent word. This paper presents an analysis of resyllabification across words in different Indian languages and its implications in Indian language text-to-speech (TTS) synthesis systems. The evidence for resyllabification is evaluated based on the acoustic analysis of a read speech corpus of the corresponding language. This study shows that the resyllabification obeys the maximum onset principle and introduces the notion of prominence resyllabification in Indian languages. This paper finds acoustic evidence for total resyllabification.

The resyllabification rules obtained are applied to TTS systems. The correctness of the rules is evaluated quantitatively by comparing the acoustic log-likelihood scores of the speech utterances with the original and resyllabified texts, and by performing a pair comparison (PC) listening test on the synthesized speech output. An improvement in the log-likelihood score with the resyllabified text is observed, and the synthesized speech with the resyllabified text is preferred 3 times over those without resyllabification.

Index Terms: resyllabification, text-to-speech systems, phonology, acoustic phonetics, derived onsets.

1. Introduction

Resyllabification is a phonological process that occurs at syllable or word boundaries where the coda of syllable or word is moved to the onset of the following syllable in continuous speech. A uniform resyllabification process observed in both stress-timed and syllable-timed languages is that the word-final coda shifts to the onset position of the next word which starts with a vowel. For example, the Hindi phrase "हम अपनी" (/ham apni/) is pronounced as "ह मप नी" (/ha.map.ni/) instead of "हम अप् नी" (/ham.ap.ni/). In this phrase, the word-final coda /m/ of the syllable हम् is shifted to the onset position of the syllable अप् of the following word. This paper investigates the phonetic and phonological evidence for resyllabification in two Indian languages - Hindi (an Indo-Aryan language) and Malayalam (a Dravidian language) and evaluates its implications on text-to-speech synthesis (TTS) systems.

The paper is organized into two parts. The first part (Section 2) describes the phonological process of resyllabification and the second part (Section 3) details the implications of the resyllabification across words in Indian language TTS systems. Section 4 concludes the work.

2. Resyllabification in Indian languages

The following section details the background work on resyllabification, the methodology adopted for the current study, resyllabification rules and related acoustic analysis performed.

2.1. Background

It was reported that resyllabification occurs in two ways such as within the domain of prosodic word and at a higher level namely prosodic phrases, intonational phrases, and utterances [1].The common resyllabification process that the pre-vocalic coda shifts to the onset of the following word is reported in Spanish [2], French [3], other languages like Turkish, Korean, Arab, Indonesian [4–6], while within word resyllabification is reported for stress-timed languages like English [7,8]. In the Indian phonology literature, most of the studies report the resyllabification process that occurs between syllables intraword rather than interword [9–11]. In [12], Maddieson discussed the phonetic cues to syllabification and resyllabification within a word in different languages, including Indian languages - Tamil, Kannada, Bengali, and Telugu. In [13], Fletcher discussed the akshara based writing system in Indian languages and the tendency of the formation of akshara units (C^*V, where C is a consonant and V is a vowel:where * represents C is optional) during syllabification. In-spite of these analyses on resyllabification within the words,specific works on phonetics and phonology of resyllabification across the word boundary not explored much in the implications of TTS. The existence of resyllabification across word boundaries in Indian English (IE) and its significance in building letter-to-sound (LTS) rules for IE is reported in [14]. In [14], Rupak showed that the quality of the synthesized speech of IE TTS system improved with the use of resyllabification rules. However, such studies have not been performed in the context of Indian language TTS systems, where LTS rules have been applied only on isolated words.

The onsets derived out of resyllabification are called derived onsets, represented as $V_1C\#V_2$ [2] (V_1 is the vowel preceding the resyllabified coda in the first word, V_2 is the vowel following the resyllabified coda in the following word, C is the derived onset which shifts from the coda position of first word, and # is the word boundary). The well-formed syllable structure that already exists as a part of syllabification in a language is called canonical onsets ($V_1\#CV_2$, where C is the canonical onset). Many of the theoretical works show that the derived onsets and

10.21437/Interspeech.2018-1176

canonical onsets share common phonetic properties. If a derived onset (word final coda) and its corresponding canonical onsets are phonetically similar, it is considered as total or complete resyllabification.

Why does Resyllabification occur? In [15] Church reported two "*tie-breaking principles*" related to resyllabification which occurs due to the insufficiency of phonotactic constraints, namely, maximum onset principle and stress resyllabification principle.

1. *Maximum onset principle (MOP):* When phonotactic and morphological constraints permit, maximize the number of consonants in onset position (e.g., re-tire).
2. *Stress resyllabification (SRP):* When phonotactic and morphological constraints permit, maximize the number of consonants in stressed syllables (e.g., re-cord vs. rec-ord)'.

The maximum onset principle is widely accepted because it is quite robust. Since the term stress creates ambiguity in syllable-timed languages, we use the term "prominence" instead of stress and we refer to this as *prominence resyllabification principle* (PRP). Even though MOP is useful, it is considered as a heuristic principle [16]. According to this study, both MOP and PRP are important for the resyllabification process in Indian languages due to various reasons. Firstly, Indian languages follow the akshara symbols which based on alphasyllabary[1] writing system and each letter represents an akshara and due to this, there is a tendency maintain more CV stricutre in speech. Hence, when a word end with a cosonant followed by a word begin with a vowel can merge each other with resyllabification to maintain the CV structure in speech. Secondly, since syllable-timed languages have a tendency to avoid complex onsets and coda compared to stress-timed languages, it is possible that the coda position gets resyllabified in some cases[2]. In Indian languages, which are syllable-timed, the syllable structure has very less complex coda compared to stress-timed languages. Thirdly, PRP principle is applicable in Indian languages. If we perceive the Hindi phrase हम आपके (/ham aapke/), it is pronounced as हमापके (/ha.maap.ke/) with more prominence on आपके (/aapke/) and due this /m/ of हम (/ham/) gets resyllabified with आप (/aap/). Finally, since Indian languages are phrasal languages [20], resyllabification is also part of creating prosodic phrases for keeping phrasal rhythm. For example Malayalam sentence: അവൻ ഓടി (/avan ootxi/) is pronounced as അവനോടി (/a.va.noo.txi/) as a single prosodic phrase due to resyllabification across noun and verb. Further, resyllabification causes phonological restructuring.

2.2. Methodology and Procedure

The analysis is limited to scripted read speech data, as the main objective is to evaluate the role of resyllabification in TTS applications. A subset of the Indic TTS database [21] is used for the analysis. The data was recorded in a high quality noise free studio environment at 48 kHz sampling rate 16-bit precision with neutral prosody, by professional native voice artists. The sentences for recording are declarative in nature and are collected from various domains - stories, news, sports etc. Around 5 hours of speech data (5000 utterances) from male speakers of Hindi and Malayalam is considered for the analysis.

Acoustic analysis was conducted using Praat speech software [22]. Duration of onsets and codas are used as the acoustic evidence for the existence of resyllabification. We compare the duration of derived onset (Vk#V) with canonical coda (Vk#C) and canonical onset environment ((C)V.kV, V#kV) (/k/ is a consonant, # is the word boundary, . is the syllable boundary, (C) are the optional consonants that could be part of a syllable). Duration is measured in Praat manually. Resyllabification is predicted if the derived onset is similar in duration and energy to canonical onset. If the derived onset is similar to the canonical coda, resyllabification is not predicted. For phonetic transcription, this study has used the common label set developed for Indic languages for TTS applications [23].

2.3. Resyllabification rules

Uniform resyllabification rule: The general resyllabification principle states that if a word ends with a closed syllable (C^*VC) and the subsequent word begin with a syllable without an onset (VC^*), the coda of the closed syllable of the first word shifts to the onset position of the word-initial syllable (syllable without an onset) of the subsequent word.
$C^*VC\#VC^* \rightarrow C^*V\#CVC^*$ (# is the word boundary).

1. *Derived onsets in Hindi:* $Vk\#V$, $Vr\#V$, $Vm\#V$ $Vn\#V$ (/k/, /r/, /m/, /n/ are consonants)
   Examples: a) धारमिक असमिता (/darmik asmita/) → धार मि कस मि ता(/dar.mi.**ka**.smi.ta/) (b) ओर अनुवाद (/our anuwad/) → ओ र नु वाद (/ou.**ra**.nu.wad/) (c)काम आगया (/kaam aagaya/) → का मा ग या (/kaa.**maa**.ga.ya/) (d) हम अपनी (/ham apnii/) → ह मप् नी (/ha.**map**.nii/).

2. *Derived onsets in Malayalam:* $Vl\#V$, $Vr\#V$, $Vm\#V$ $Vn\#V$ (/l/, /r/, /m/, /n/ are consonants) Malayalam is considered as a *no coda* or vowel ending language and only word-final coda possible is *chillaksharam* [3]. Unlike a consonant represented by an ordinary consonant letter, this consonant is never followed by an inherent vowel. Other consonants end with a halant (eu)[4]. This kind of diacritic is common in Indic scripts, generically called virama in Sanskrit, or halant in Hindi..
   Examples: a) ഹൃദയാഘാതം ഉണ്ടാകുന്നത് (/hridyaaghaadam undakunnat/) → / hridyaaghaada**mu**ndakunnatu/(mu), b) സൈന്യാധിപൻ ഉൾപ്പടെയുള്ള (/sainyaadhipan ulxppetxeyulxlxa/) → സൈന്യാധിപനുൾപ്പടെയുള്ള (/sainyaadhipan**ul**xppetxeyulxlxa/)

[1] The term alphasyllabary is coined by Bright in [17].

[2] "phonological analysis could be based on comparing syllable complexity, an approach that has been used since Roach (in [18]) suggested that stress-timed languages allow complex onsets and codas, but syllable-timed languages tend to avoid them or impose lower limits on the length of consonant clusters than stress-timed languages" ([19]).

[3] A chillu, or a chillaksharam is a special consonant letter that represents a pure consonant independently, without help of a virama.

[4] Chandrakkala (candrakkala) is a diacritic attached to a consonant letter to show that the consonant is not followed by an inherent vowel or any other vowel (for example, ka → k).

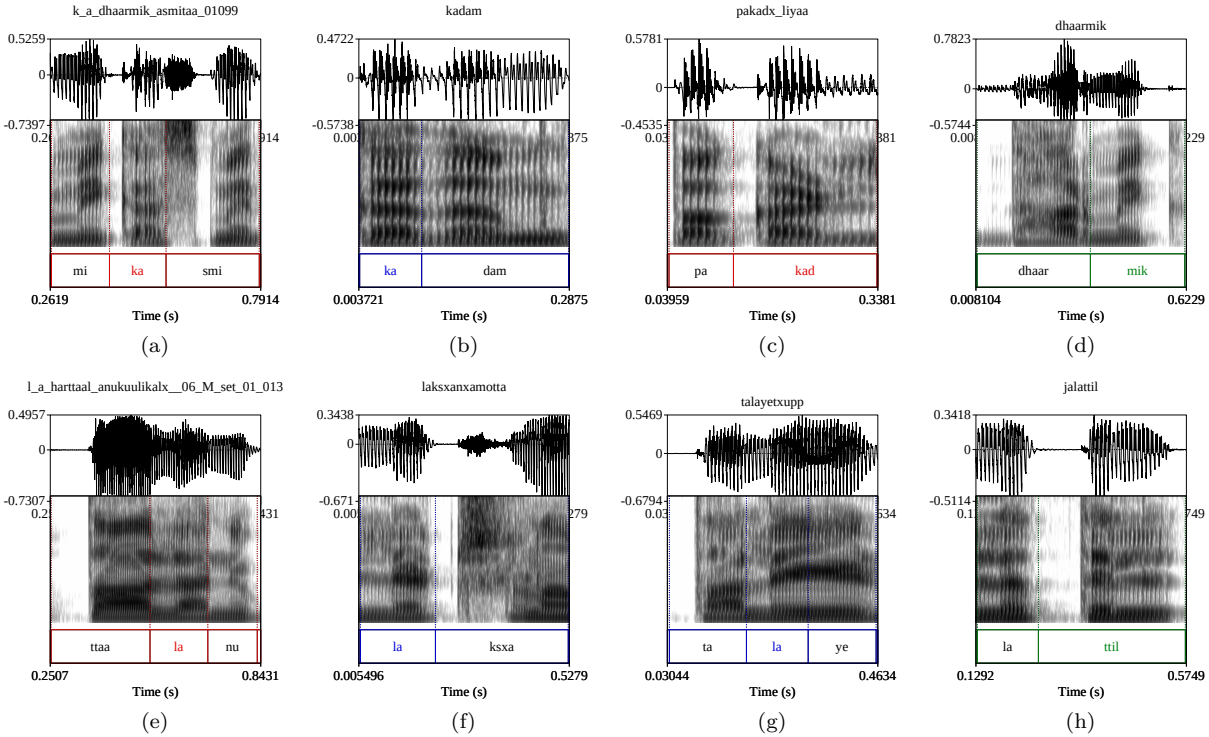

Figure 1: *Spectrogram and waveform of derived onset (red), canonical onsets word initial and word medial (blue),and canonical coda (green) for a Hindi phone /k/ (Figures 1(a)-1(d)), and for a Malayalam phone /l/ (Figures 1(e)-1(h))*

Language specific rule for Malayalam for geminates ending syllable: In Malayalam when a geminate consonant acts as the onset final syllable of a word and the next word starts with a vowel, the virama ◌് (*eu*) associated with the geminate consonant is deleted and the geminate is resyllabified with the initial vowel of following word due to vowel sandhi.
$C^*VC^*.CCV\#VC \rightarrow C^*VC^*\#CCVC^*$
Derived geminate onset in Malayalam: $Vkk\#V$, $V\#pp$, $V\#tt$, $V\#txtx$ (/kk/, /pp/, /tt/, /txtx/ are geminate consonants)
Example: ആവശ്യക്കാർക്ക് ഉപയോഗിക്കാനും (/aawashyakkaarkkeu upayoogikkaanuq/) → ആവശ്യക്കാർക്കുപയോഗിക്കാനും (/aawashyakkaarkkupayoogikkaanuq/)

2.4. Results

This study finds that all derived onsets stand with relatively equal duration and similar intensity with its corresponding canonical onsets in word-initial and word-medial syllable positions. It is interesting to note that derived onsets and the corresponding canonical coda differ in the acoustic properties with more duration and less energy for canonical codas. Table 1 provide the duration of derived onsets in comparison with canonical onsets and canonical coda. Figures 1(a) - 1(d) provide the acoustic analysis for /k/ in Hindi and Figures 1(e) - 1(h) show the acoustic analysis for /l/ in Malayalam. All other derived onsets in this study provide the same analysis. Here duration and energy are used as the primary acoustic correlates to determine resyllabification. The resyllabification of geminated consonant in Malayalam also shows the derived onsets are similar to canonical onsets. Since there is no canonical coda, as Malayalam is a no-coda language, the comparison with canonical coda cannot be performed. Since both derived and canonical onset show similar characteristics, it is considered as total resyllabification.

Table 1: *Mean duration of resyllabified derived onsets (V#CV versus canonical onsets #CV. ,V.#CV and canonical coda VC#*

	V#CV		#CV.		V.#CV		VC#	
	Hindi	Malay-alam	Hindi	Malay-alam	Hindi	Malay-alam	Hindi	Malay-alam
/m/	0.06	0.04	0.07	0.05	0.06	0.05	0.12	0.1
/n/	0.04	0.05	0.04	0.07	0.07	0.06	0.1	0.13
/l/	0.05	0.06	0.06	0.05	0.04	0.05	0.12	0.1

3. Text-to-speech synthesis systems

Text-to-speech synthesis (TTS) systems convert given input text to corresponding speech output. The process of building a TTS system involves several steps - collection of training data (text sentences and corresponding speech utterances), applying letter-to-sound rules (converting text sentences to the sequence of sub-word units, syllables and phones), speech segmentation (finding sub-word unit boundaries in speech utterances), prosody modeling, training, and testing. During training, the sub-word unit boundary information obtained after parsing and segmentation, along with other features[5] is used to train TTS systems. During synthesis, the input text is parsed into sub-word units, and are given to the synthesizer.

[5] Other features include spectral and excitation (fundamental frequency) features is used to train TTS systems. Most widely used spectral feature is mel generalized cepstral feature [24].

The performance of the synthesised output is evaluated in terms of naturalness and intelligibility..

3.1. Letter-to-sound (LTS) rules

LTS plays an important role in deriving the pronunciation for the given text. LTS rules are applied in the state-of-the-art Indian language TTS systems using a unified parser for Indian languages [25]. The unified parser applies LTS rules on isolated words after tokenizing text sentences. However, resyllabification across words happen quite often in natural speech as discussed in Section 2. The state-of-the-art parsers fail to capture such resyllabification rules. To account for the same, the rules enunciated in Section 2 are applied to the text before passing the word to the unified parser.

The resyllabification rules are not applied across words belonging to different phrases that are separated by a pause.

3.2. Evaluation

Two evaluation metrics are used for comparison, namely (1) log-likelihood scores of the syllable/phone boundaries after speech segmentation[6] and (2) pair comparison (PC) test on the synthesized speech obtained from both systems.

3.2.1. Log-likelihood score

The acoustic log-likelihood score of a phone belonging to a segment of speech is computed after performing forced Viterbi alignment (FVA) after speech segmentation during training. HTK toolkit was used for performing FVA [27]. Log probability of the entire utterance is computed by accumulating the log-likelihood of the constituent phones as given in Equation 1:

$$logP(\mathcal{O}|\lambda) = \sum_{i=1}^{N} logP(x_i|\lambda_i) \quad (1)$$

$\mathcal{O}$ in Equation 1 refers to acoustic features of an entire speech utterance, x_i are a set of feature vectors of individual phones, λ_i is the corresponding individual HMM phone model, λ is the entire set of concatenated HMMs, and N is the number of phones in the utterance. Mel frequency cepstral coefficients (MFCC) features are the acoustic features used. GMMs are used for acoustic modeling. 5-state 2-mixtures, 3-state 2-mixtures, and 1-state 2-mixtures are used for phone modeling for vowels, consonants, and silences respectively. Log-likelihood scores are computed for the sequence of phones with resyllabification and without resyllabification using equation 1. Table 2 shows the average log-likelihood scores of speech utterances with and without application of resyllabification rules. It is observed that the average log-likelihood score improves with resyllabification rules.

3.2.2. PC listening test

PC listening test was conducted on the synthesized speech utterances from the systems with and without resyllabification. HMM-STRAIGHT based TTS systems [28] are used for speech synthesis. In this test, the same sentence synthesized with two systems are played to the listener and the listener is asked to give a preference. The sentences are randomly ordered to avoid bias. The order independent preference percentage of each system is obtained from the number of times the samples from each system is preferred independent of the order. The listeners are also allowed to choose equal preference if they are not able to distinguish between the two. The result of the PC test is shown in Figure 2. The test is conducted on 18 sentences with 20 participants each. It is observed that around 60% (i.e 3 times more than) the system with resyllabification is preferred over that without resyllabification around 15%. This shows that incorporating resyllabification rules improves the quality of the synthesized speech.

[6] Hybrid segmentation (HS) algorithm is used for segmenting speech data [26].

Table 2: *Average log-likelihood scores of utterances with and without the application of resyllabification rules*

Language	Gender	Average log-likelihood without resyllabification	Average log-likelihood with resyllabification
Hindi	male	-8544.46	-8228.01
Malayalam	male	-3348.21	-3346.86

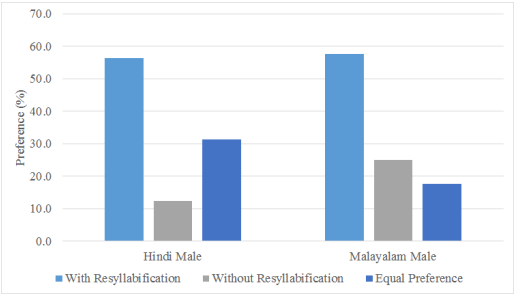

Figure 2: *Results of pairwise comparison tests: Order independent preference for systems with and without resyllabification*

4. Conclusions

This paper analyzed possible resyllabification rules in two Indian languages based on the acoustic analysis of the read speech corpora of a subset of Indic TTS database. Our analysis shows that the resyllabification happens even in read speech and hence it is relevant for preserving naturalness of speech. The resyllabification rules are applied in the text processing component in the TTS systems. Both quantitative and qualitative evaluation of the built system shows that the goodness of the TTS system improved after applying resyllabification rules.

5. Acknowledgements

The datasets used in this work are part of the project "Development of Text-to-Speech Synthesis for Indian Languages Phase II". The authors thank the Department of Information Technology, Ministry of Communication and Technology, Government of India for the same. The authors also thank anonymous reviewers for their comments.

6. References

[1] M. Nespor and I. Vogel, *Prosodic phonology: with a new foreword*. Walter de Gruyter, 2007, vol. 28.

[2] P. Strycharczuk and M. Kohlberger, "Resyllabification reconsidered: On the durational properties of word-final/s/in spanish," *Laboratory Phonology: Journal of the Association for Laboratory Phonology*, vol. 7, no. 1, pp. 1–24, 2016.

[3] C. Fougeron, O. Bagou, M. Stefanuto, and U. H. Frauenfelder, "Looking for acoustic cues of resyllabification in french," pp. 2257–2260, 2003.

[4] J. Kornfilt, *Turkish*. Routledge, 2013.

[5] L. Brown and J. Yeon, *The handbook of Korean linguistics*. John Wiley & Sons, 2015.

[6] K. de Jong, K. Okamura, and B.-j. Lim, "The phonetics of resyllabification in english and arabic speech," in *Proc. 15th ICPhS Barcelona*, 2003, pp. 2621–2624.

[7] P. Kiparsky, "Metrical structure assignment is cyclic," *Linguistic inquiry*, vol. 10, no. 3, pp. 421–441, 1979.

[8] W. Labov, "Resyllabification," *Amsterdam studies in the theory and history of linguistic science series 4*, pp. 145–180, 1997.

[9] P. Pandey, "Akshara-to-sound rules for hindi," *Writing Systems Research*, vol. 6, no. 1, pp. 54–72, 2014.

[10] T. Mohanan, "Syllable structure in malayalam," *Linguistic Inquiry*, pp. 589–625, 1989.

[11] S. Nag, "Learning to read alphasyllabaries," *Theories of reading development*, pp. 75–87, 2017.

[12] I. Maddieson, "Phonetic cues to syllabification," pp. 203–217, 1985.

[13] C. M. Fletcher-Flinn, "Frontiers in the acquisition of literacy," *Frontiers in psychology*, vol. 6, pp. 1019–1020, 2015.

[14] S. Rupak Vignesh, A. Shanmugam, and H. Murthy, "Significance of pseudo-syllables in building better acoustic models for indian english tts," pp. 5620–5624, 03 2016.

[15] K. W. Church, "Phonological parsing in speech recognition," pp. 133–138, 1987.

[16] B. Hayes, *Introductory phonology*. John Wiley & Sons, 2011, vol. 32.

[17] W. Bright, "A matter of typology: Alphasyllabaries and abugidas," pp. 63–70, 2000.

[18] P. Roach, "On the distinction between 'stress-timed'and 'syllable-timed'languages," *Linguistic controversies*, vol. 73, p. 79, 1982.

[19] R. Fuchs, "Speech rhythm in varieties of english," in *Speech Rhythm in Varieties of English*. Springer, 2016, pp. 87–102.

[20] C. Féry, "The intonation of indian languages: An areal phenomenon," *Problematizing Language Studies. Festschrift for Ramakant Agnihotri. New Delhi: Aakar Books*, pp. 288–312, 2010.

[21] A. Baby, A. L. Thomas, N. L. Nishanthi, and T. Consortium, "Resources for indian languages," in *CBBLR – Community-Based Building of Language Resources*. Brno, Czech Republic: Tribun EU, Sep 2016, pp. 37–43.

[22] P. Boersma, "Praat: doing phonetics by computer," *http://www.praat.org/*, 2006.

[23] B. Ramani, S. L. Christina, G. A. Rachel, V. S. Solomi, M. K. Nandwana, A. Prakash, S. A. Shanmugam, R. Krishnan, S. K. Prahalad, K. Samudravijaya *et al.*, "A common attribute based unified hts framework for speech synthesis in indian languages," in *Eighth ISCA Workshop on Speech Synthesis*, 2013, pp. 311–316.

[24] K. Tokuda, T. Kobayashi, T. Masuko, and S. Imai, "Mel-generalized cepstral analysis-a unified approach to speech spectral estimation," in *Third International Conference on Spoken Language Processing*, 1994, pp. 279–280.

[25] A. Baby, N. L. Nishanthi, A. L. Thomas, and H. A. Murthy, "A unified parser for developing Indian language text to speech synthesizers," in *International Conference on Text, Speech and Dialogue*, Sep 2016, pp. 514–521.

[26] S. A. Shanmugam, "A hybrid approach to segmentation of speech using signal processing cues and Hidden Markov Models," M. S. Thesis, Department of Computer Science Engineering, IIT Madras, India, July 2015. [Online]. Available: "http://lantana.tenet.res.in/thesis.php"

[27] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey *et al.*, "The htk book," *Cambridge university engineering department, New Jersy*, 2006.

[28] H. Kawahara, I. Masuda-Katsuse, and A. de Cheveigne, "Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds," *Speech Communication*, vol. 27, no. 3–4, pp. 187 – 207, 1999.