Pioneering Dzongkha Text To Speech Synthesis

Department Of Information and Technology, Bhutan.
NECTEC, Thailand.

Overview

- Introduction
- Development
  - Phoneme Design for Dzongkha TTS
  - TTS Design and development
- Evaluation and discussion
- Future prospects
- Conclusion

Introduction

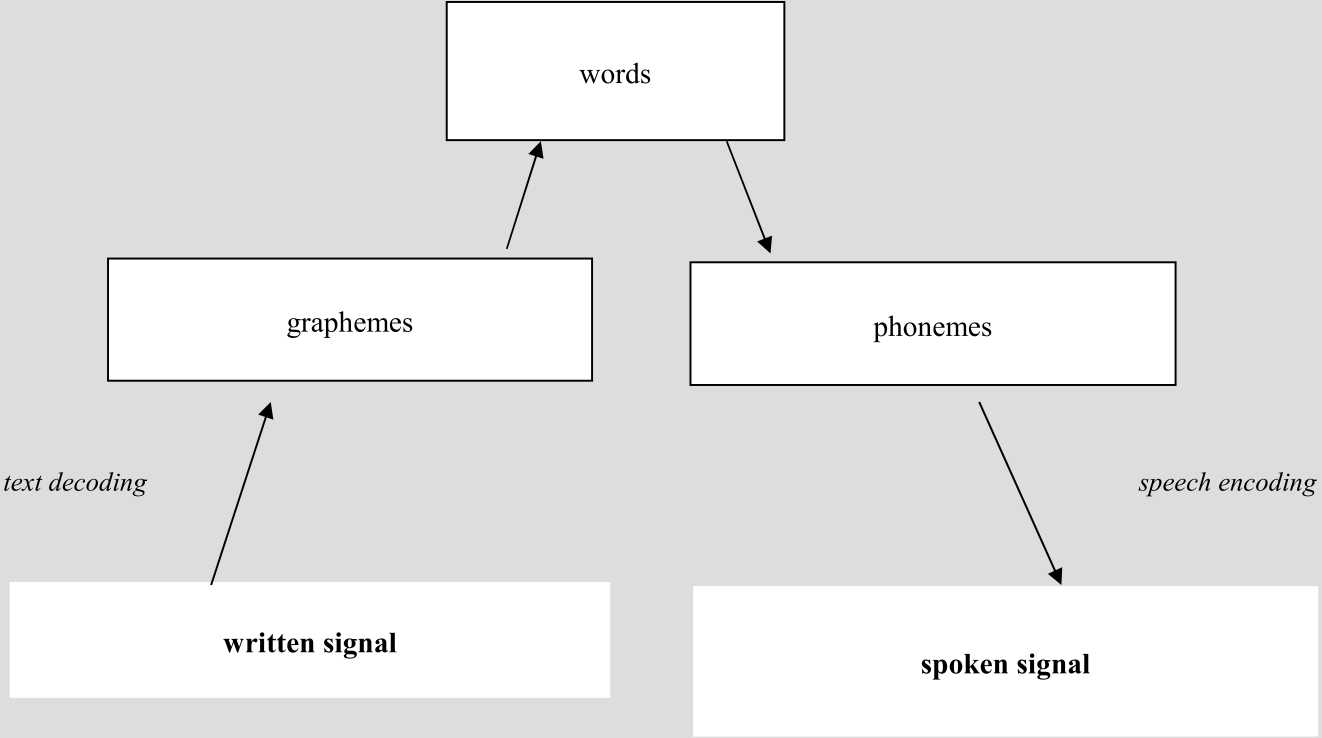
- It consisted of
  - designing a phoneme set
  - building a text processor
  - designing and collecting speech database
  - training HMM under HMM-based speech synthesis system (HTS) toolkit
  - integrating all components in an application.

Introduction(contd.)

- The key features of the TTS
  - Text analysis
  - Speech synthesis
  - Figure 1 below shows these two features
  - Text analysis finds intermediate forms (Syllables in case of Dzongkha TTS)
  - Synthesizing generates speech signals from that intermediate form.

Introduction(contd.)

- Figure 1: The common form model of TTS (P. Taylor. 2008)

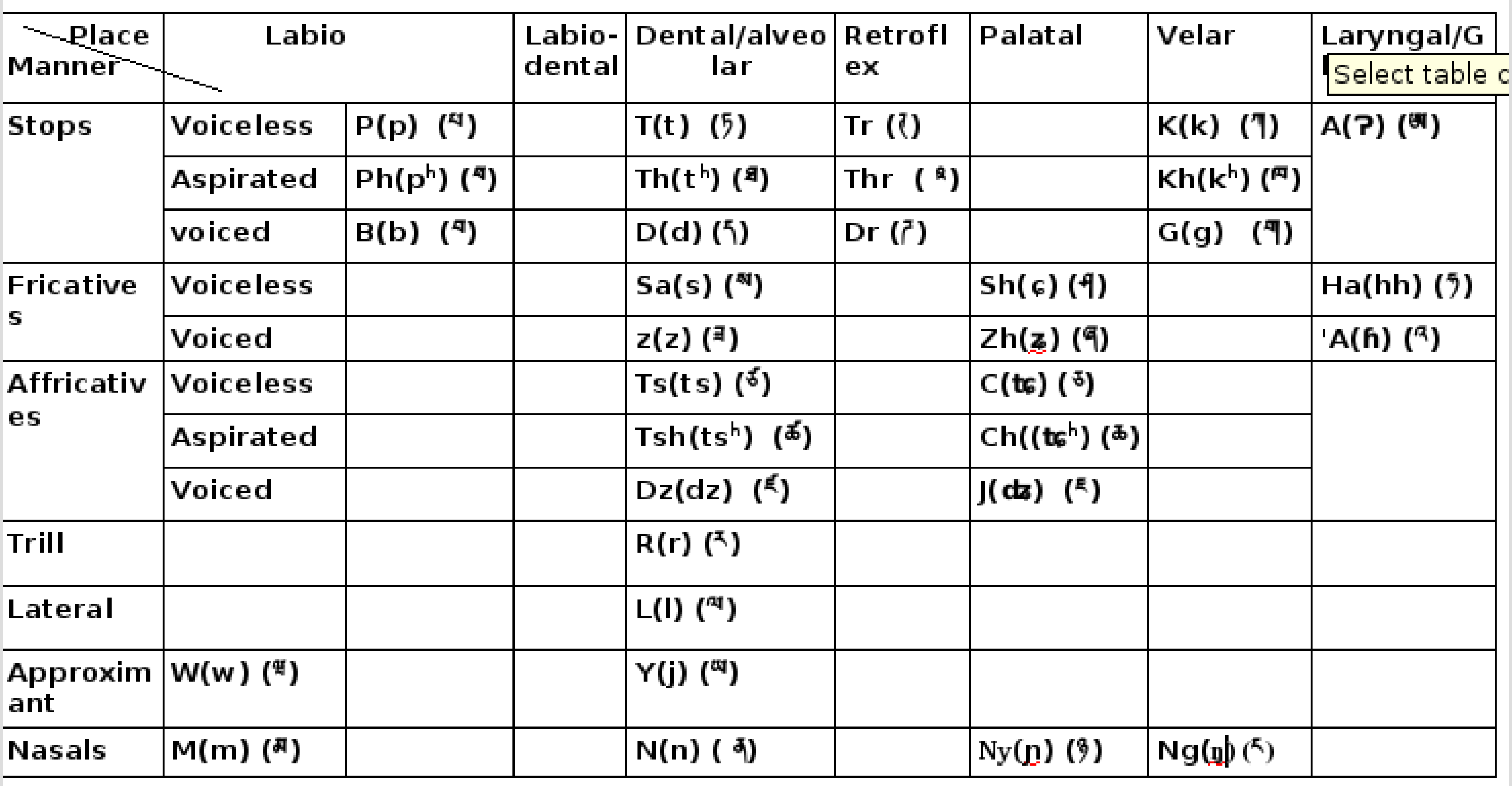


Introduction(contd)

- Dzongkha TTS
  - HMM-based
    - Uses accostics paramaters to generate speech
    - These are synthesized from context dependent HMM-models
  - HTS version 2.0
  - MCEP (mel-cepstral coefficients)
  - Log F0
  - Duration parameters

Dzongkha sound system

- Figure 4(a): Dzongkha IPA table(consonants)

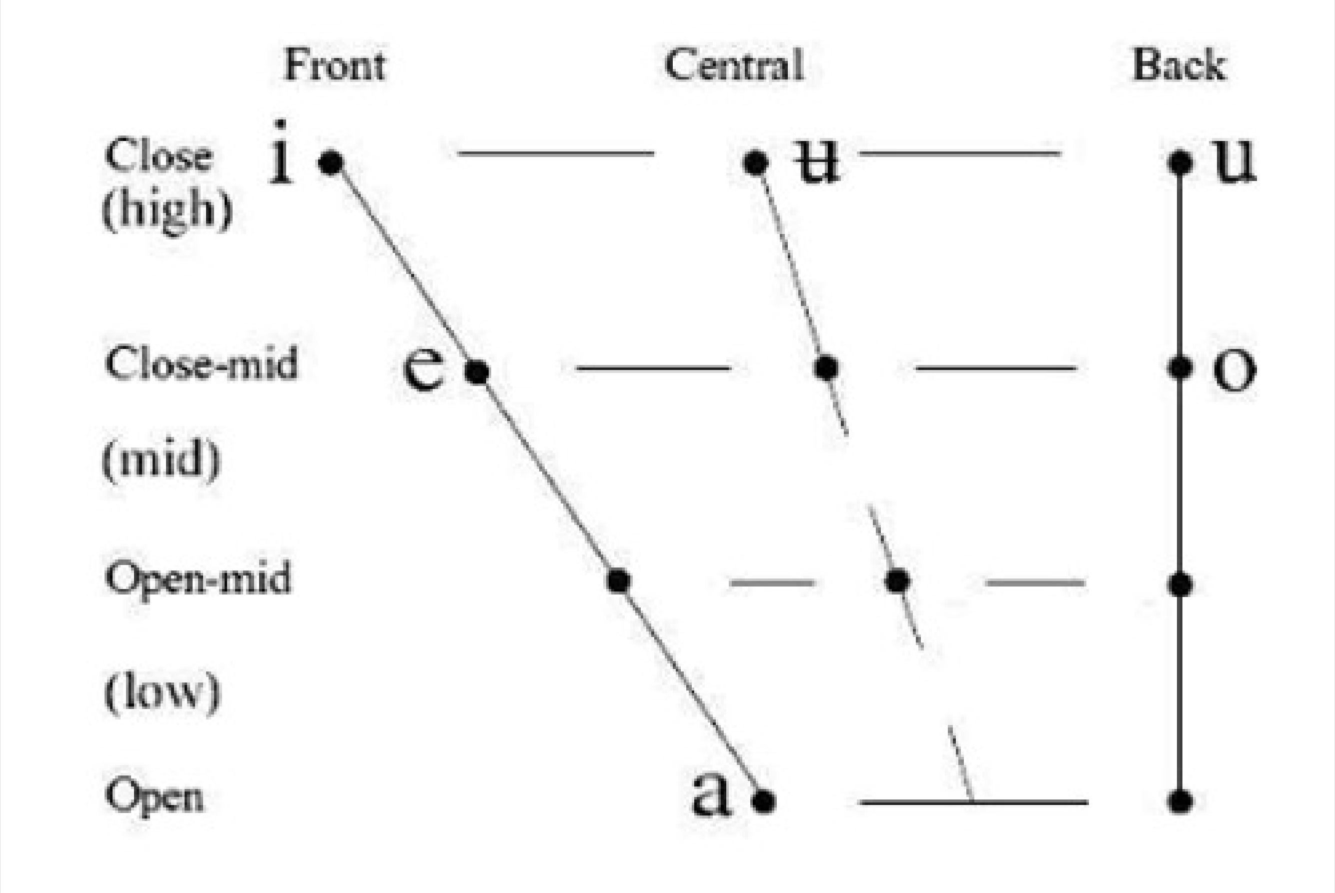


Place Manner	Labio		Labio-dental	Dental/alveolar	Retroflex	Palatal	Velar	Laryngal/G
Stops	Voiceless	P(p) (པ)		T(t) (ཏ)	Tr (ཊ)		K(k) (ཀ)	A(ʔ) (ཨ)
	Aspirated	Ph(p^h) (ཕ)		Th(t^h) (ཐ)	Thr (ཋ)		Kh(k^h) (ཁ)	
	voiced	B(b) (བ)		D(d) (ད)	Dr (ཌ)		G(g) (ག)	
Fricatives	Voiceless			Sa(s) (ས)		Sh(ɕ) (ཤ)		Ha(hh) (ཧ)
	Voiced			z(z) (ཟ)		Zh(ʑ) (ཞ)		'A(ɦ) (འ)
Affricatives	Voiceless			Ts(ts) (ཙ)		C(tɕ) (ཅ)		
	Aspirated			Tsh(ts^h) (ཚ)		Ch((tɕ^h) (ཆ)		
	Voiced			Dz(dz) (ཛ)		J(dʑ) (ཇ)		
Trill				R(r) (ར)				
Lateral				L(l) (ལ)				
Approximant	W(w) (ཝ)			Y(j) (ཡ)				
Nasals	M(m) (མ)			N(n) (ན)		Ny(ɲ) (ཉ)	Ng(ŋ) (ང)	

Select table c

Dzongkha sound system

- Figure 4(b): IPA table for Dzongkha (vowels)



Dzongkha sound system

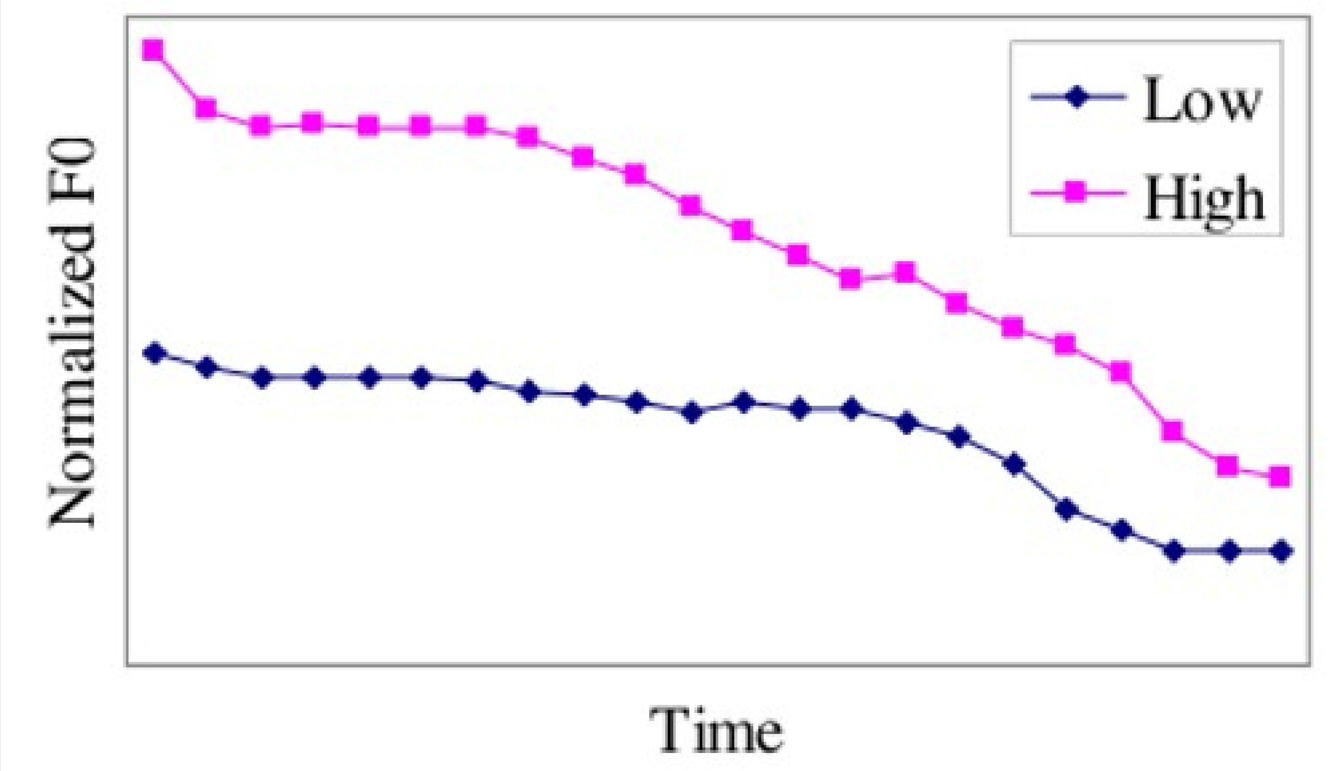
- Representation of spoken Dzongkha
  - Initial consonants
  - Consonant clusters with single consonants
  - Vowels
  - Diphthongs
  - An inherent vowel 'a' is always present with single consonants
  - Some vowels are modified when root letter combines with certain suffices
  - Clusters in Dzongkha is represented by stacking root letter over the subjoined letter

Dzongkha sound system

- Dzongkha tone system
  - Two tone system
  - The low tone normally used
  - The high which is the modification of the low tone
  - Modification depends on combination of certain prefixes ('ག' 'ད' 'བ' 'མ'), head letter ('ས' 'ར') and subjoined ('ལ') with root letter.

Dzongkha sound system

- Figure 5: Normalized F0 contour of the syllable 'lam' showing high tone (meaning monk) and low tone (meaning road or way)

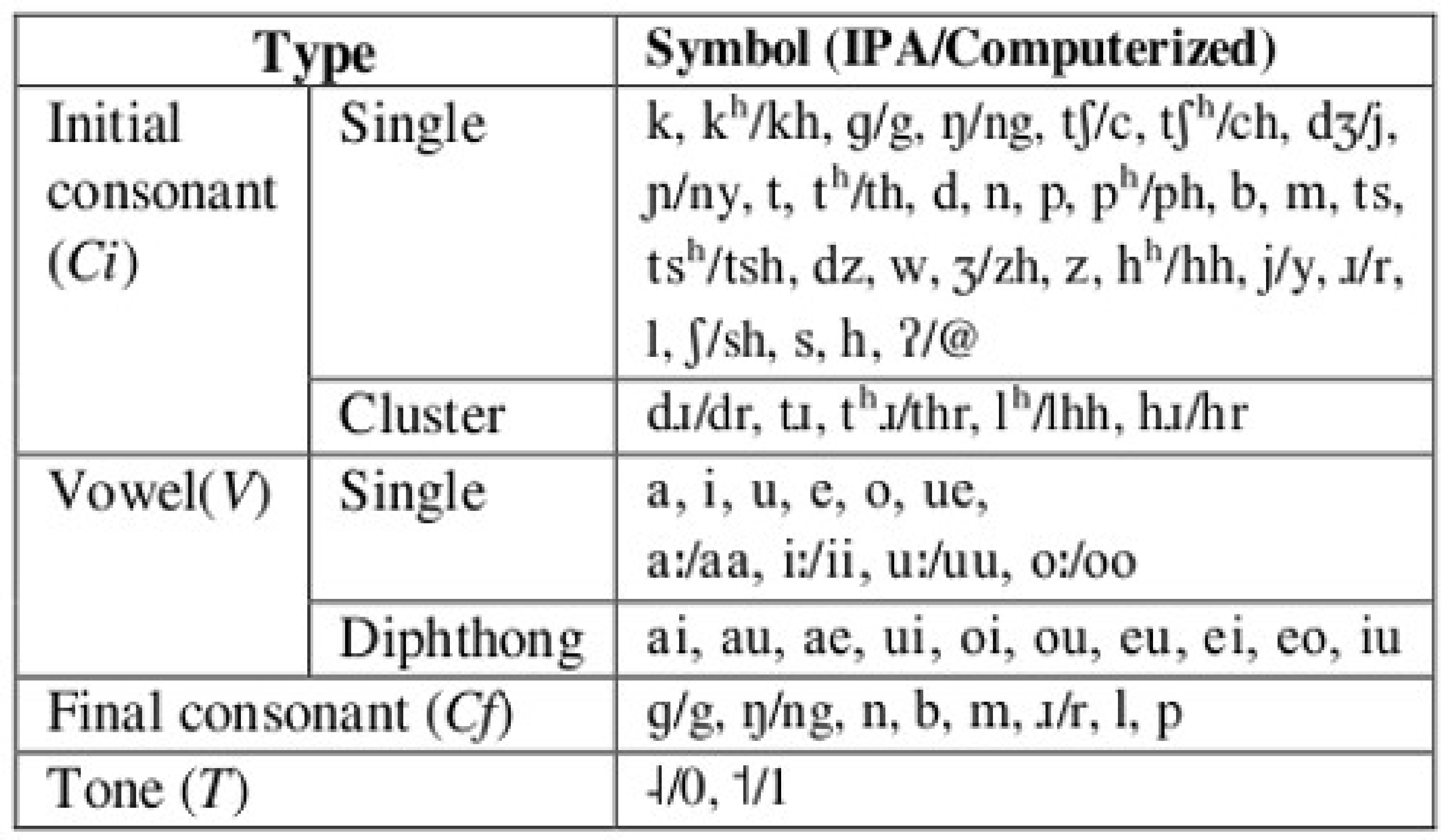


Phoneme design for Dzongkha TTS

- Observations during transcription
  - 30 initial consonants, 5 initial consonant clusters, 10 vowels and 10 dipthongs defined
  - single phonemes from figure 4 were employed
  - Four more vowels were observed and defined separately ('aa','ii','uu','oo').
  - Consonant clusters mostly formed by combination with 'r' sound
  - Some suffices are not pronounced ('d' 's' 'hh')
  - Certain suffices modifies the vowel

Phoneme design for TTS

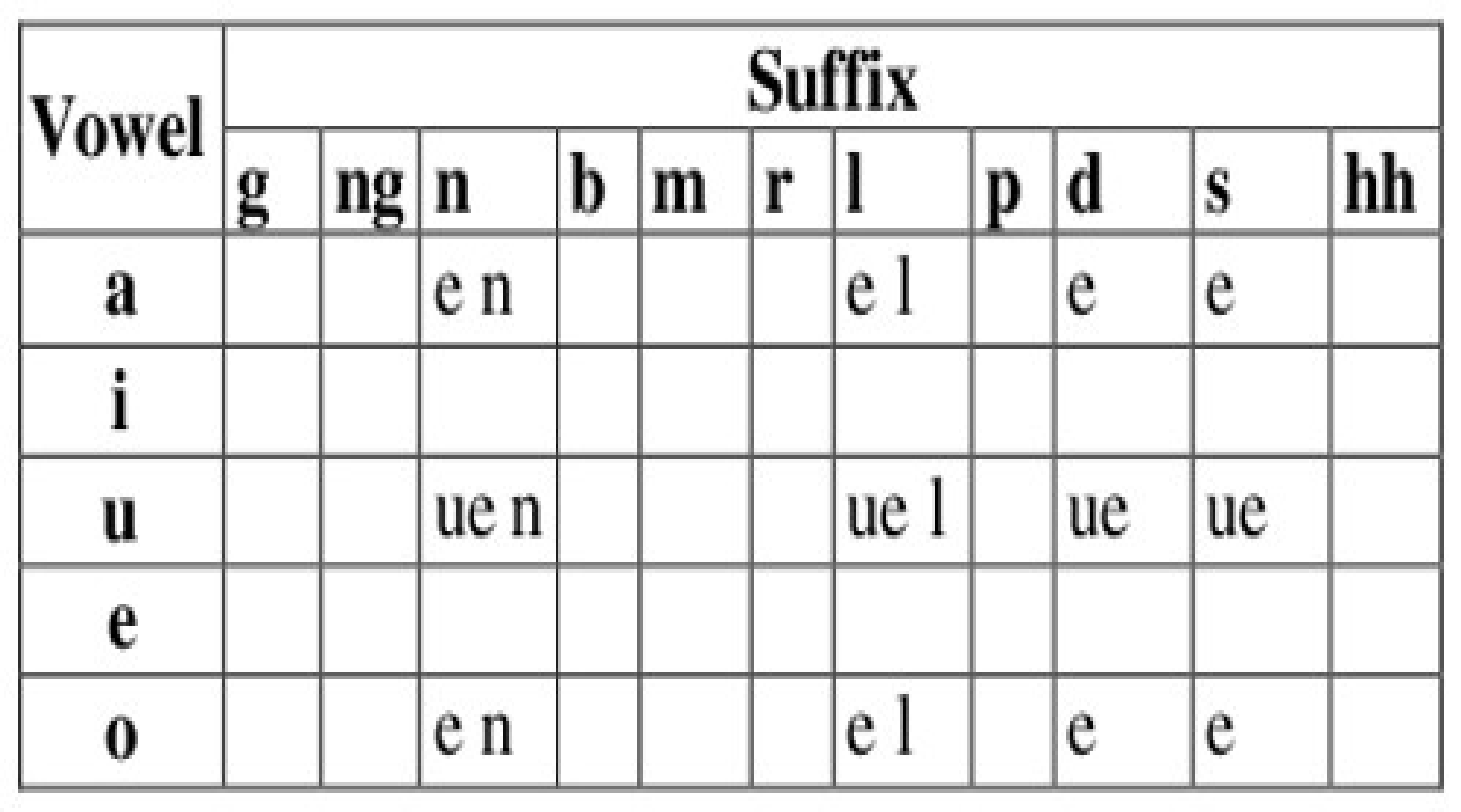
- Table 1: Dzongkha phoneme inventory for TTS



Type		Symbol (IPA/Computerized)
Initial consonant (*Ci*)	Single	k, kʰ/kh, g/g, ŋ/ng, tʃ/c, tʃʰ/ch, dʒ/j, ɲ/ny, t, tʰ/th, d, n, p, pʰ/ph, b, m, ts, tsʰ/tsh, dz, w, ʒ/zh, z, hʰ/hh, j/y, ɹ/r, l, ʃ/sh, s, h, ʔ/@
	Cluster	dɹ/dr, tɹ, tʰɹ/thr, lʰ/lhh, hɹ/hr
Vowel(*V*)	Single	a, i, u, e, o, ue, a:/aa, i:/ii, u:/uu, o:/oo
	Diphthong	ai, au, ae, ui, oi, ou, eu, ei, eo, iu
Final consonant (*Cf*)		g/g, ŋ/ng, n, b, m, ɹ/r, l, p
Tone (*T*)		˨/0, ˦/1

Phoneme design for TTS

- Vowel modification with suffices.
- Table 2: Modification of vowel.



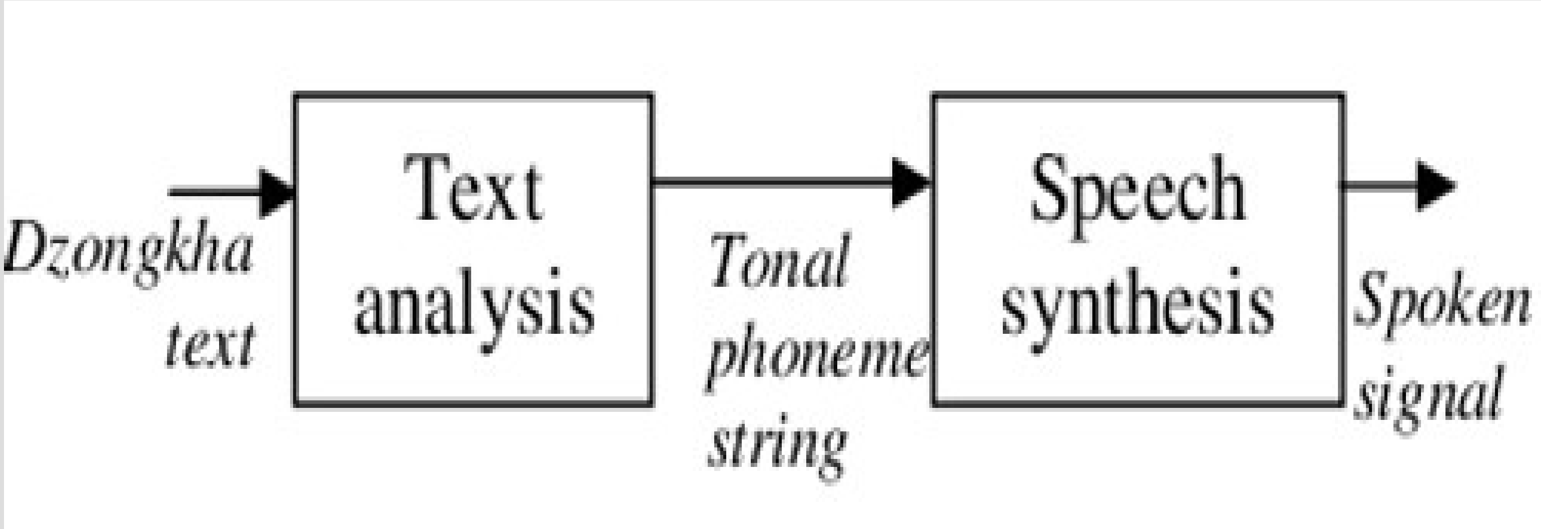
Vowel	Suffix										
	g	ng	n	b	m	r	l	p	d	s	hh
a			e n				e l		e	e	
i											
u			ue n				ue l		ue	ue	
e											
o			e n				e l		e	e	

Phoneme design for TTS

- Tonal representation
  - Digit symbol '0' for low
  - Digit symbol '1' for high

TTS design and development

- The system consists of two main modules, text analysis and speech synthesis.
- Figure 6: The proposed system structure.



TTS design and development

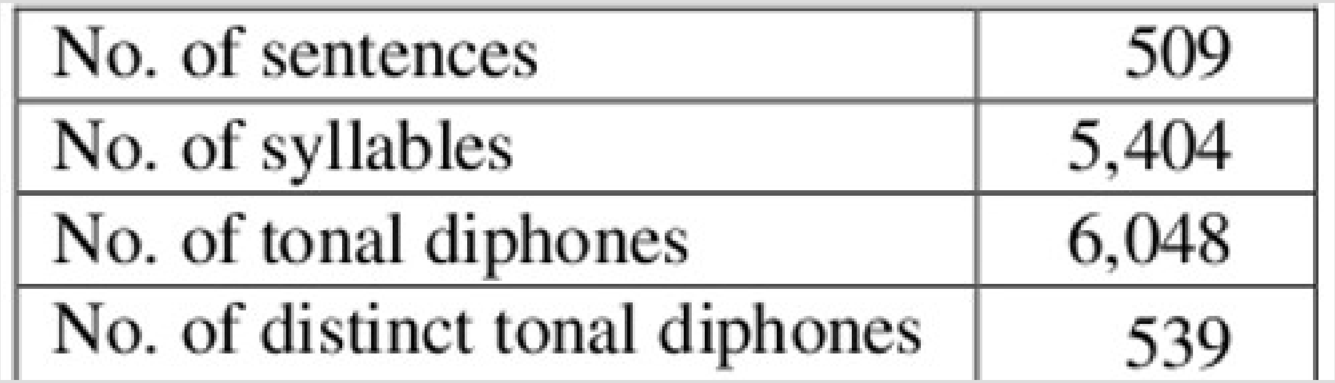
- Text analysis
  - Implemented using a dictionary based G2P
  - Presence of a syllable marker makes it easier to implement G2P using a look up dictionary
- Dictionary
  - A Dzongkha text corpus of 40,000 sentences were collected
  - Top 4000 distinct syllables occurring were included in the dictionary

TTS design and development

- Speech synthesis
  - A corpus of 509 sentence included
  - These had to cover all 53 phonemes including two tones shown above in Table 1.
- The sentence selection
  - Iteratively select a sentence with most distinctive tonal di-phones
  - Stop when all tonal di-phones in text corpus are included

TTS design and development

- Table 3: Dzongkha speech corpus statistics.



No. of sentences	509
No. of syllables	5,404
No. of tonal diphones	6,048
No. of distinct tonal diphones	539

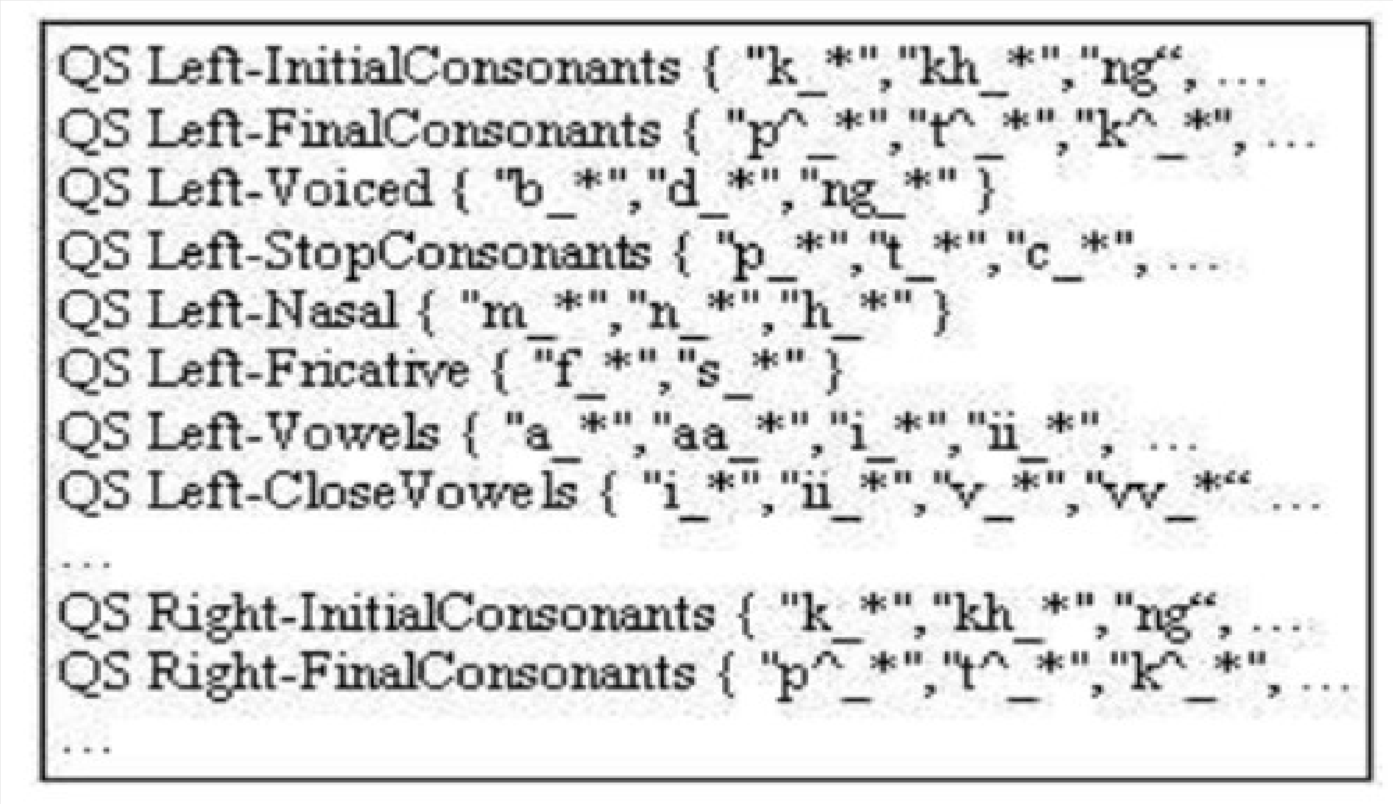
TTS design and development

- Building synthesizer
  - Mel-Cestrum (MCEP), duration and Log fundamental frequency (Log F0) were extracted from each utterance in the speech corpus
  - By using HTS with HTK and SPTK HMMs can trained in a flat start manner
  - It doesn't require any phoneme boundary tag but only phoneme transcription of each utterances

TTS design and development

- A clustering tree designed for Dzongkha phoneme is used in HMM state tying.

- Figure 7: A part of clustering tree used for HMM state tying.



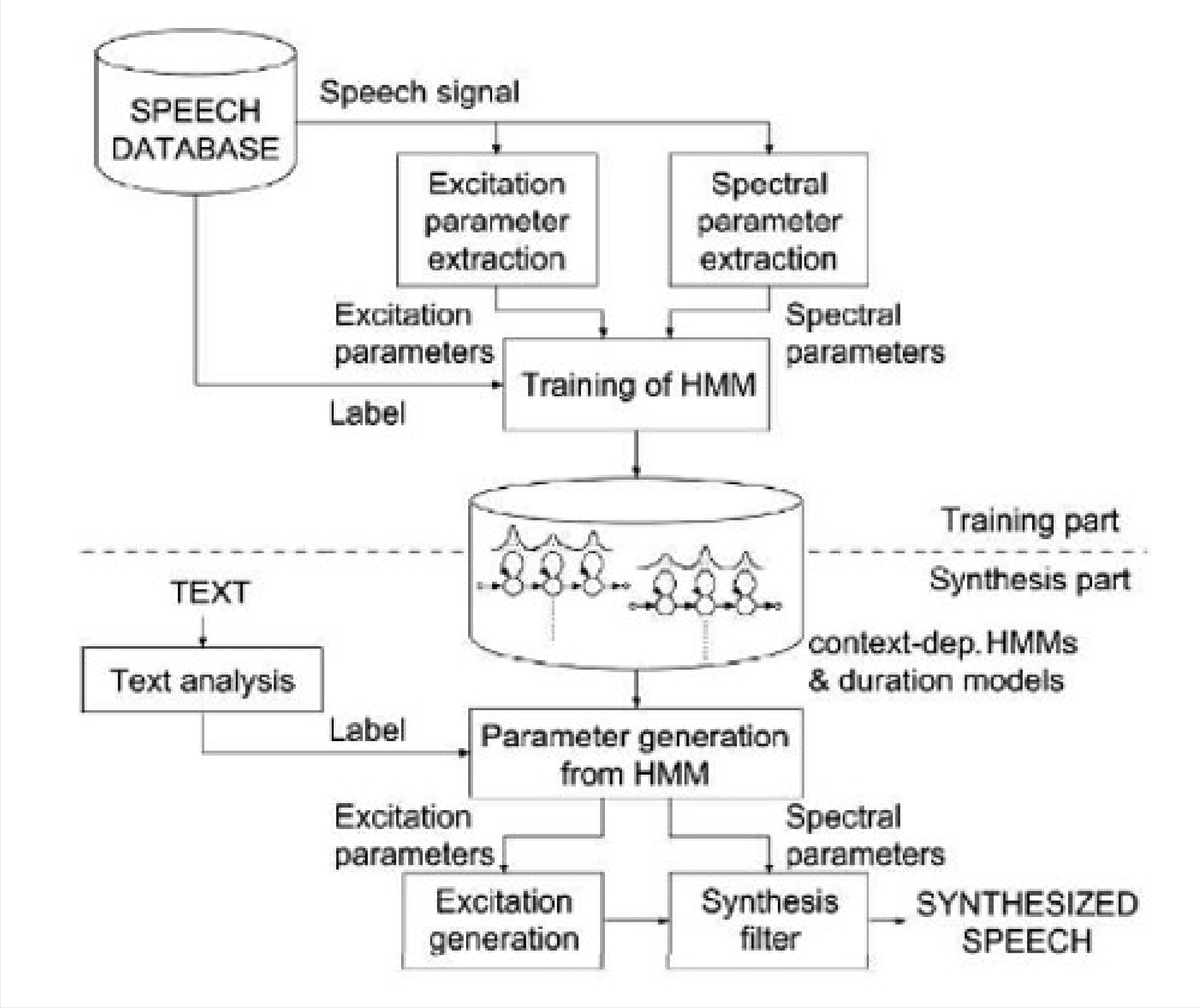
```
QS Left-InitialConsonants { "k_*","kh_*","ng", ...
QS Left-FinalConsonants { "p^_*","t^_*","k^_*", ...
QS Left-Voiced { "b_*","d_*","ng_*" }
QS Left-StopConsonants { "p_*","t_*","c_*", ...
QS Left-Nasal { "m_*","n_*","h_*" }
QS Left-Fricative { "f_*","s_*" }
QS Left-Vowels { "a_*","aa_*","i_*","ii_*", ...
QS Left-CloseVowels { "i_*","ii_*","v_*","vv_*" ...
...
QS Right-InitialConsonants { "k_*","kh_*","ng", ...
QS Right-FinalConsonants { "p^_*","t^_*","k^_*", ...
...
```

TTS design and development

- Building the synthesizer
  - Using the training script in the HTS, the HMMs are trained to construct the synthesizer
  - Given trained HMMs, the “hts-engine” command with the toolkit could be evoked to synthesize speech.

TTS design and development

- Figure 8: HTS toolkit usage.



Evaluation and discussion

- Evaluation
  - based on mean opinion score
  - Fifteen Bhutanese were asked to evaluate
- Score system
  - 1 to 5
  - 1 for worst
  - 5 for best
- Result
  - human speech rated 3.93
  - synthesized rated 3.19

Future prospects

- Enlarging speech corpus with larger di-phone coverage
- More distinct syllables required by the G2P module
- Important prosody generation modules
  - pausing between words and phrases
  - duration and F0 modeling

Conclusion

- Building the first Dzongkha TTS
  - designing a phoneme inventory
  - building a text processor
  - designing and creating the speech database
  - training HMMs under HTS frame work
  - integrating all these into an application
- Yet more work needs to be done to improve speech quality as mentioned in future prospects