Dzongkha Text-to-Speech Synthesis System – Phase II

Dechen Chhoeden, Chungku
Department of Information Technology
and Telecom , Bhutan
{dchhoeden, chungku}@ dit.gov.bt

**Ananlada Chotimongkol,
Anocha Rugchatjaroen, Ausdang
Thangthai, Chai Wutiwiwatchai**
HLT Laboratory
National Electronics and Computer
Technology Center, Thailand
{ananlada.chotimongkol, anocha.rug,
ausdang.tha, chai.wut} @nectec.or.th

Abstract

This paper describes the development of advanced Dzongkha text-to-speech (TTS) system which is a marked improvement over the first Dzongkha TTS prototype (Sherpa et al., 2008), using the Hidden Markov Model-based speech synthesis (HTS) method. Advanced Natural Language Processing techniques like word segmentation and phrase boundary prediction were integrated with the earlier prototype to improve the quality of the synthesized speech. These advanced techniques and the integration procedure are explained in this paper. With the inclusion of these advanced modules, we could improve the quality of the synthesized speech as measured by a subjective listening test, Mean Opinion Score (MOS), from 2.41 to 2.98. The procedure of the integration is explained in this paper.

1. Introduction

The initial work on TTS focused more on the linguistic aspect. In that, defining a phonetic set for the language, a grapheme-to-phoneme conversion module using a simple phoneme look-up dictionary, text normalization, and collection of speech corpus were some of the main natural language processing (NLP) tasks that were explored. All these modules were put together with a simple text processing model and a speech synthesizer trained using the HTS framework. As the prototype included segmentation only at syllabic level and as word segmentation and phrase boundary prediction which are integral components of natural speech were not taken into account, the output speech sounded robotic as the prosodic information was limited to just the syllable boundaries.

In the second phase of Dzongkha TTS development, advanced NLP techniques have been incorporated to improve the quality of the synthesized speech. This is done by adding a prosody generation module for proper phone duration and pausing which requires development of many sophisticated NLP algorithms including a word segmentation algorithm, a Part-Of-Speech (POS) tagger, a phone duration predicting algorithm and a phrase boundary predictor. These algorithms require a deeper understanding of the language in order to annotate both speech and text corpora with POS tags, word and phrase boundaries in order to train various statistical models.

1.2. Structure of Dzongkha Text

Dzongkha language belongs to the Sino-Tibetan family of languages. The language structure is not very complicated. It follows the subject-object-verb order of sentence structure like most Asian languages. The smallest unit of text that is meaningful is the 'syllable'. Dzongkha text has syllabic and sentence boundaries; however, there are no spaces between words. A syllable is separated from another syllable by a character ' ་ ' called 'Tsheg' but we do not have word boundaries that separates words from each other. A word is at least one syllable long. As such there are no separate punctuation marks to differentiate the words from one another.

བཞེས་སྒྲོ་བཟའ་ཚུགས།

zhe-go-za-tshu
“can eat”

Figure 1. Example Sentence

Similarly, though the sentences are clearly delimited by sentence boundaries ('།'), no explicit rules are in place for having phrase boundaries. Spaces inserted between phrases depend on the writer or the speaker. It is necessary for the long sentences to be divided into a number of shorter utterances for the listener or the speaker to be able to understand or read the sentences, respectively. In the case of speech, the boundaries to be inserted can be short pauses between the utterances.

The objective of any TTS system is to emulate natural speech as much as possible. To do that the inclusion of NLP processes of word segmentation and phrase boundary prediction becomes imperative. Hence these processes are integrated in the second phase of Dzongkha TTS development to improve the system.

2. Design and Development

In phase-I of Dzongkha TTS development, the text analysis was completed to convert the text to intermediate forms of syllables and attach their corresponding phonemic representation. And the speech synthesizer would generate speech using HTS method. This model was chosen owing to its small footprint, stability, smoothness and speaker adaptability (Sherpa et al., 2008). Using HMM method, spoken signal is generated from acoustic parameters that are synthesized from context dependent HMM models. In phase-II the same process is followed with the integration of additional NLP modules.

In phase-II, to improve the synthesized speech quality, we have increased the size of the speech database used for training the speech synthesizer. We have also incorporated a prosody generation module to the system to make the synthesized speech more natural with more appropriate phone duration and pausing. The following diagram (Figure 2) illustrates the process that was used to improve the quality of the synthesized speech in the new system.

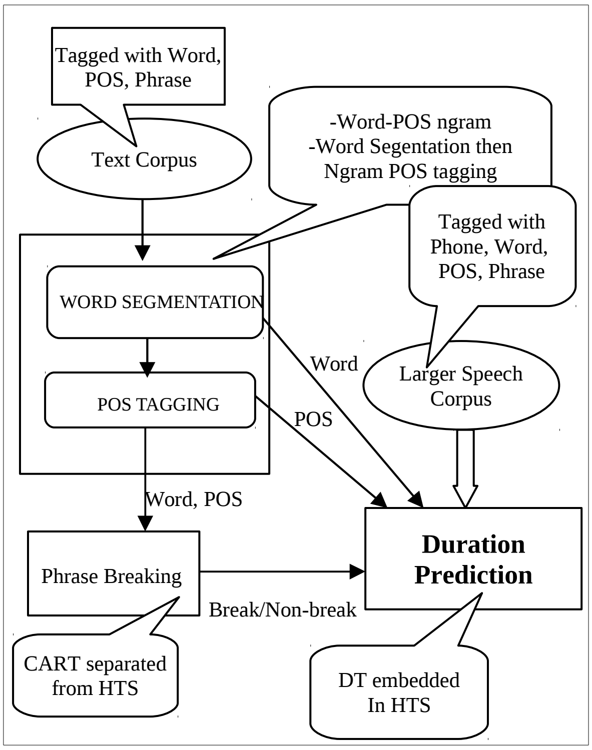

Figure 2. The development process of the Dzongkha TTS system Version II (Training phase)

From the diagram, we can see that word boundary and POS are crucial information for predicting a phrase boundary. Phrase boundaries together with word boundaries and POS tags are, in turn, crucial information for predicting phone duration. Hence, in order to predict phrase boundaries and phone duration in the prosodic generation module, a word segmentation module and a POS tagger are necessary components for extracting features during the executing phase of the TTS system. The training phase of the Dzongkha TTS system version 2 comprises of 5 steps, data preparation and the development of 4 necessary components: word segmentation, POS tagging, phrase boundary prediction and phone duration prediction.

2.1. Data Preparation

When building a TTS system a number of different types of data needs to be prepared. Text corpus and Speech corpus being the major portion of data, apart from that, data like POS-annotated text are also required. The following describes the data that were prepared:

Text Corpus: Dzongkha Corpus is a collection of text with 600,000 syllables (400,000 words approximately) from different genres like newspaper articles, samples from traditional books, novels, dictionary. A small part of this corpus which contains approximately 40,277 words, was annotated with word boundaries, phrase boundaries and POS tag of each word. These features are necessary for training a phrase boundary prediction and a phone duration prediction modules.

Larger speech corpus: 434 additional sentences were recorded to increase the speech corpus to **943** speech utterances. These sentences covering all the phones in the language, were selected from the text corpus with the selection criteria described by Wutiwiwatchai et al. (2007) and Sherpa et al. (2008). A female native speaker recorded the sentences and the resulting wave files were converted to a 44 Khz and 16 bits format. The speaker was the only one available in that particular circumstance.

Increase the syllable entries in the Pronunciation dictionary: About 900 new syllables were transcribed with their respective pronunciation (a sequence of phones). Transcription of syllables involves marking each phone in the syllable with special symbols borrowed from the IPA system.

> For example, the syllable 'ཁབ' is transcribed as '**kh-a-b-0**', which consists of three phonemes (initial consonant, vowel and final consonant) and '**0**' indicates normal tone. High tone is represented by '**1**'.

Additional features of speech: Word and phrase boundaries were added using special symbols in each sentence (**934** sentences from the speech corpus) and each of the words in those sentences were tagged with their respective POS tags. This is a crucial step in our TTS building phase as this data will be used in the “labelling” process of HTS to train HTS model.

Example sentence: མདའ་བརྐྱབ་ད་མདའ་གཉིསཔ་ཆ་རང་བཀག་རེག་ཡོག་ནི།

Annotation: d-a-x-0|*/NN/c-a-b-0|*/VBAt/d-a-x-0|*/TM/d-a-x-0|*/NN/ny-i-p-1|ch-a-x-0|*/NN/r-a-ng-0|*/CC/k-a-x-0|r-e-x-0|*/NN/ph-o-x-0|n-i-x-0|*/VB/$-$-$-$|*/ /
i

'*' indicates the word boundary while “**$-$-$-$|***” is used to represent phrase boundaries.

Preparation of label files for the new sentences: Preparation of label files with word boundary, phrase boundary and POS information. The data from the above step will be used to prepare label files for the “labelling” process.

2.2. Advanced NLP modules

Word Segmentation

Segmentation of words is basically the process of tokenizing a string of text into words. Many different methods are available for word segmentation nowadays. The accuracy of each of the methods will differ from language to language. Hence a language needs to adopt the best algorithm that gives the best accuracy. For Dzongkha TTS we have adopted the “longest string matching” method for segmenting words, as it is suitable for a small corpus as is our case. From the limited amount of training data, we used this technique which is a dictionary-based method of word segmentation. It depends on a match between word entries in the dictionary and a string of characters in an input text, and segments the text accordingly. Given the input text the algorithm scans the characters from left to right and finds the longest matching entry in the dictionary.
There are many draw-backs with this method. For one there is the problem of unknown word or out-of-vocabulary (OOV) word where some words may not match any word in the dictionary. Secondly, there may be more than one ways to segment the input sequence of characters. These drawbacks can be improved with the adoption of algorithms with better accuracy.

POS Tagging

An integral part of prosody is the POS of the words. An annotated text with POS information for each word, is required to add additional functionality to the TTS system, to support vital features of natural speech. An automatic POS tagger (Chungku, et al., 2010) was used to tag text corpus for phrase boundary prediction.

Example of Dzongkha POS-tagger output text:

Input Text	Output Text	
སྤུ་རོ་	སྤུ་རོ་	NNP
རིན་སྤུང་རྫོང་	རིན་སྤུང་རྫོང་	NNP
༡༩༨༨	༡༩༨༨	ND
ལུ་	ལུ་	PP
།	།	PUN

Table 1. Output of POS Tagger

Also the 934 sentence utterances had to be manually POS-tagged along with the word and phrase boundary annotations for training HTS, as explained in Section 2.1.
Example of POS Annotation of raw text:

Raw text: འབྲུག་དང་བརྟ་དེཤ་གཉིས་ཀྱི་བར་ན་ ཟུང་འབྲེལ།

POS Annotated equivalent: འབྲུག་ /NNP དང་/CC བར་ དེཤ་ /NNP

གཉིས་ /CD ཀྱི་/CG ཟུང་འབྲེལ་/NN །/PUN

Phrase boundary prediction

Phrases are also an integral part of prosody in speech. A Phrase boundary predictor is necessary in a TTS system for the synthesized speech to sound natural. It will break down a long utterance into meaningful parts. In Dzongkha there are no explicit phrase boundaries, although boundaries at syllabic and sentence levels are present. At every sentence end, it is certain that a speaker must pause and then start reading the next sentence, however it is unsure of the pattern of phrases and pauses within a sentence. In Dzongkha speech, phrase breaks can occur between words, sentences, numbers, spaces, punctuation and symbols.

To predict phrase breaks in a Dzongkha sentence, we use a machine learning algorithm called Classification And Regression Tree (CART). CART is a learning algorithm that is based on a binary decision tree and has been applied widely in the task of phrase boundary prediction (Hansakunbuntheung et al., 2005). Its ability to manipulate both symbolic and real-value features and the ability to handle sparse training data are the advantages of this model.

We use POS information, features at word and syllable levels to train a CART tree to predict whether to insert a pause at the current location or juncture with respect to each word. These features are described in more detail in the following table. The last two rows in the table are the output of the CART decision tree.

Feature	Description
POSL2,POSL1,POSR1, POSR2	POS with respect to the position of the current juncture. For e.g, POSL1 is the POS of the word one place on the left hand side of the current juncture. POSR1 is the POS of the word one place to the right with respect to the position of the current juncture. [each]
CurrSylInPhr	No. of Syllables between the current juncture and the beginning of the current sentence.
CurrWrdInPhr	No. of Words between the current juncture and the beginning of the current sentence.
WrdInPhr	Total no. of words in the current sentence.
SylInPhr	Total no. of syllables in the current sentence.
NB	Non-break for junctures that is not a phrase break.
B	Break for junctures that is a phrase break.

Table 2. Required Features for each word to train CART

For training CART, the data-set or corpus without removing any punctuation, symbols and numbers were used considering the fact that in real life all kinds of text-data will be present. The required features are extracted from this corpus to train CART, as a result a '*decision tree'* is formed which can then be used to automatically predict phrase boundaries for a given input text. It basically looks at the Break (B) or Non-break (NB) status of each word in the training corpus, and puts these information in the decision tree along with other features. In the executing phase of the TTS, the 'B' and 'NB' features of an input string will be output by the “phrase prediction” process, and this output file will be used in the “labelling” process for training HTS to help produce synthesized speech containing silences wherever the phrase breaks were predicted.

Phone duration prediction

To predict the duration of each phone, features like word boundaries, phrase boundaries and POS of each word are necessary, together with phone duration from natural speech. To do that we needed to manually tag the word boundaries, POS tags and the phrase boundaries of the 943 sentence utterances from the speech corpus. In the HTS training process these information will be utilized to generate a phone duration model.

2.3. HTS Training Process

As in phase I of Dzongkha TTS development, in phase-II as well, HTS speech engine is used as the speech synthesizer of the system. Here the naturalness is improved by adding more context to each phone label for creating duration trees and using more appropriate context-based HMMs. The extra features of POS, word and phrase boundaries, position in word and phrase, are added to the question file (clustering tree) for training additional phrase boundary prediction module. After the training process is successfully completed, HMMs and tree files which are needed for the HTS synthesizer are created. To synthesize a speech file, the synthesizer also needs a context-based phone label file of target speech, generated during the execution process of the system.

2.4. Execution phase: Putting it all together

In the execution phase (Figure 3) all the modules that were developed are integrated together with the HTS, using a command prompt application. To explain the execution process, given a text input, the string of text is first segmented into words. These words are then tagged with their corresponding POS tags. Numbers and dates are normalized into letter-form and the phoneme string for each word is also looked-up using the syllable dictionary (Sherpa, et al., 2008). Phrase breaking process then predicts phrase breaks and non-breaks. After these steps we have an output in the format as shown below. This output file has word entries (from the input text). In that each word has its corresponding POS information, break or non-break situation, and the transcription (letter to sound) of each word along with the tone (0 or 1) of the word.

Word	*POS*	*B/NB*	*Transcription*
ཉིནམ་	NN	NB	ny-i-m-0|
ཉཔ	NN	NB	ny-i-x-1|p-a-x-0|
ལྟག་ལ་ཁ་	NN	NB	t-a-x-0|l-a-x-0|kh-a-x-0|
བྱག་ཕྱུག་ན	NN	B	j-a-x-0|ph-u-x-0|n-a-x-0|

Table 3: Output file having features of a given text

This output file is then used to obtain a context label file for that particular input string. That is then used to generate the synthesized speech using the HTS synthesizer that was previously trained in the training process.

In this way the TTS Version-2 is capable of producing synthesized speech from a string of input text. The quality of the speech is discussed in the following section.

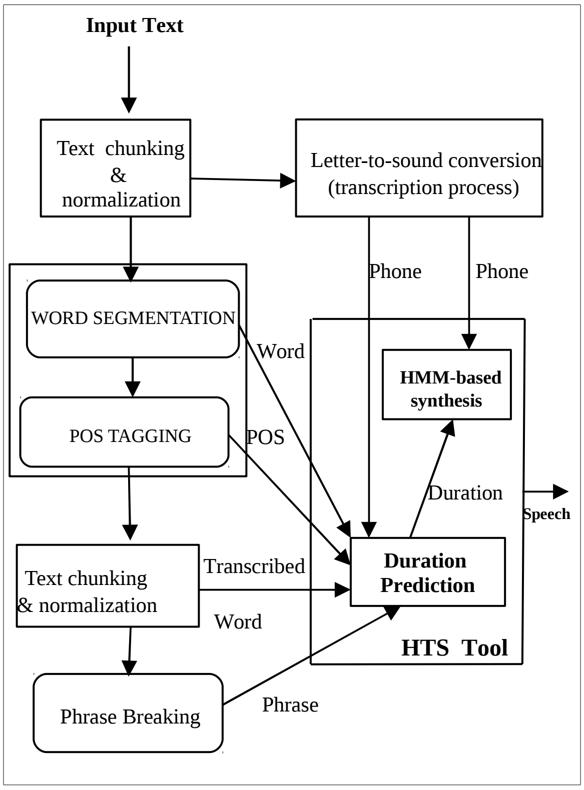

Figure 3. A diagram describing the Execution Phase of the TTS system

3. Evaluation results and Discussions

3.1. Word Segmentation

F-measure was used to evaluate the overall performance of the word segmentation algorithm. The F-measure is calculated using the equations as given below.

F-measure = **2 (Precision X Recall)/(Precision + Recall)**	**Corr** = The number of words in the output that is correctly segmented; **RefWord** = The total number of words in the reference; **OutputWord** = The total number of words in the output.
Precision = **Corr / OutputWord**	
Recall = **Corr / RefWord**	

Table 4. Equations for finding accuracy of the longest matching algorithm

To evaluate the performance of the longest matching algorithm using Dzongkha text, a dictionary was created from all unique words in the text corpus. The dictionary contains 30,856 words. The test set is the transcript of 943 wave files in the speech corpus also described in Section 2.1. The test set contains 8,701 words. The result from the evaluation is presented in the following table.

Precision	86.73
Recall	84.67
F-measure	**85.69**

Table 5. Accuracy of the longest matching algorithm

Many different word-segmentation algorithms can be explored to achieve better accuracy for word-segmentation, in future.

3.2. POS Tagging

For the POS tagger, a training data of 20,000 words which were manually tagged with their respective POS tags. This corpus was used to create a working POS tagger. 5000 words were automatically tagged with this tagger. The output from the tagger was manually corrected and found that the accuracy of the tagger was around 80%. This initial tagger was used in the development of the Dzongkha TTS system version-2.

3.3. Phrase boundary prediction

In training CART, 90% of the corpus was used as training data while 10 % of the corpus was used as test data. The training data (train.txt) is a dataset of 37537 words (9022 phrases and 2743 sentences) and the test data (test.txt) is a training data subset containing 3719 words. The performance of a phrase boundary predictor is measured in terms of percentage of correctly predicted breaks and non-breaks, similar to the criteria discussed by Hansakunbuntheung et al. (2005). An accuracy of 88.42% was achieved using precision and recall.

3.3. Evaluation of the system

The system was evaluated using Mean Opinion Scoring method (Viswanathan, 2005). Fifteen native speakers were requested to rate the new synthesized speech (*syn II*), the natural speech and the synthesized speech (*syn I*) produced by the first TTS system, based on the quality and naturalness of the speech samples. The ratings of 1 to 5, 1 being the worst and 5 being the best, were used for the fifteen speech samples which were jumbled between natural and synthesized samples. After evaluation, recorded speech had an average rating of 2.41 for "*syn I*" and 2.98 for "*syn II*". While the natural speech, understandably, had the highest average score of 4.63, the resulting MOS scores clearly indicate that the new system has improved considerably in comparison to the previous system.

The resulting speech from the system has vastly improved compared to the almost robot like speech output by the earlier system. Still the system has room for improvement by increasing both the text and speech corpus and improving the accuracy of the different modules presented in the paper.

4. Conclusion

This paper presented the integration of advanced NLP modules in Dzongkh TTS system, which included word segmentation, phrase boundary prediction, POS tagging and phone duration prediction. As a result the system has improved. This means that the synthesized speech produced by the system is closer to spoken speech containing word boundaries and some pauses in between phrases within the sentences. The evaluation score based on a subjective listening test was indicative of the fact that the system has improved, yet, it is far from being as good as the natural speech. Future work such as improving the accuracy of the integrated modules, and increasing both the text and speech corpus may help improve the system furthermore.

Acknowledgement

Firstly we would like to thank the donors, the PAN Localization project Secretariat for their continuous support. Again phase-II of Dzongkha TTS would not have progressed as it has without the kind helping hand extended by our friends from Human Language Technology (HLT) lab at NECTEC. We are very much grateful for their kind support and contribution towards the development of the advanced TTS system.

References

Chai Wutiwiwatchai, Anocha Rugchatjaroen, and Sittipong Saychum. 2007. An Intensive Design of a Thai Speech Synthesis Corpus. *The Seventh International Symposium on Natural Language Processing*. Thailand.

Chatchawarn Hansakunbuntheung, Ausdang Thangthai, Chai Wutiwiwatchai, and Rungkarn Siricharoenchai. 2005. Learning methods and features for corpus-based phrase break prediction on Thai. *European Conference on Speech Communication and Technology* (*EUROSPEECH*), pp. 1969-1972.

Choochart Haruechaiyasak, Sarawoot Kongyoung, and Matthew N. Dailey. 2008. A Comparative Study on Thai Word Segmentation Approaches. *In Proceedings of ECTI-CON*. Krabi, Thailand.

Chungku, Gertrud Faaß, and Jurmey Rabgay. 2010. Building NLP resources for Dzongkha: A tagset and a tagged corpus,. *Proceedings of the Eighth Workshop of Asian Language Resources* (*WS1*), 23rd *International Conference on Linguistics*(*Coling* 2010), 103-110. Beijing, China,.

Dechen Chhoeden, Chungku, Anocha Rugchatjaroen, Ausdang Thangthai, Chai Wutiwiwatchai, and Ananlada Chotimongkol. 2010. Technical report on Dzongkha Speech Synthesis System – Phase II.

Helmut Schmid. 1994. Probabilistic Part-of-Speech Tagging Using Decision Trees. *Proceedings of the International Conference on New Methods in Language Processing*, 44-49. Manchester, UK.

Mahesh Viswanathan. 2005. Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (MOS) scale. *Computer, Speech and Language*, volume 19, 55-83.

Uden Sherpa, Dawa Pemo, Dechen Chhoeden, Anocha Rugchatjaroen, Ausdang Thangthai, and Chai Wutiwiwatchai. 2008. Pioneering Dzongkha text-to-speech synthesis. *Proceedings of the Oriental COCOSDA*, 150–154. Kyoto, Japan.