# Report on using merlin

——Zhengkun Mei task: build arctic voices using Merlin recipe with World Vocoder

#### Basic Introduction on Merlin:

The Merlin Speech Synthesis Toolkit is used for neural network-based speech synthesis. The system takes speech features as input, uses a neural network to predict acoustic features, and then passes the acoustic features to a voice synthesizer (vocoder) to produce a speech waveform. Merlin is not a complete TTS system, it only provides the core acoustic modeling module of TTS (acoustic and speech feature normalization, neural network acoustic model training and generation). It must be used in combination with a front-end text processor (e.g. Festival) and a vocoder (e.g. STRAIGHT or WORLD).

#### Design of Merlin:

###### Installation:

I would like to describe the structure of Merlin based on the command I use and the output screenshot I made.

Firstly we download Merlin.

Second we use the following commands:

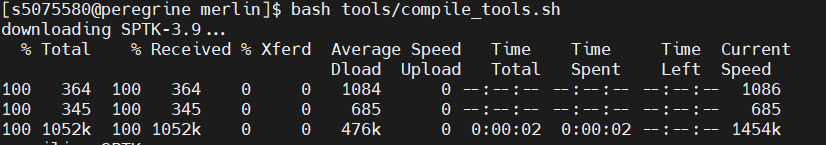
cd merlin/

bash tools/compile\_tools.sh #SPTK, WORLD

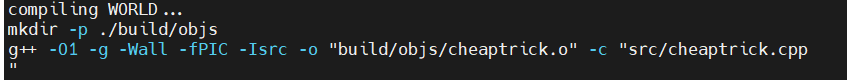
bash tools/compile\_other\_speech\_tools.sh #speech tools, festival and festvox

bash tools/compile\_htk.sh #HTK

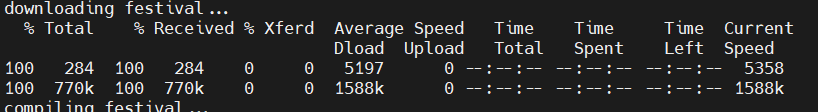
in order to install auxiliary software (frontend: festival, festvox, hts, htk; vocoder :WORLD, SPTK)



This is SPTK.



This is installing WORLD.



This is Festival. Just a few examples, I will not show all the pictures.

Then we have the Theano as the model training tool.

###### Run the Demo and Vocode Arctic Voice:

The demo uses the US English Women (slt\_arctic\_full\_data.zip) broadcasters' 1132 texts and voices from the CMU ARCTIC database, including 16kHzd waveform voices, and all the labeling is done by festvox or hts labeling scripts.

Since Merlin does not have its own frontend, the demo directly uses the label file converted by frontend as input data. The script 01\_setup.sh was used to create experiments and experiments/slt\_arctic\_full in the current folder, and the downloaded and decompressed data were placed in the following directories for the duration model and acoustic model training respectively.

For my task of building the arctic voice using vocoder, the process is:

Using the dataset to create the config file, global one, acoustic and duration config file.

Train the duration model and acoustic model based on the conf file.

Synthesis voice based on the model.

###### Conclude the structure and workflow of Merlin based on the findings:

For the training process:

For wav. File:

We extract the acoustic features and make cmp, composed acoustic feature.

For txt:

Make utt by adding full context training labels and then make labels, HTS full and monophone labels and do alignment on the full context and use HTS tool to generate duration label.

We have these two as the input to train the duration and acoustic model.

For synthesizing process:

Input the txt, give the labels, input the lab data to duration and acoustic model, get composed acoustic feature, turn them to acoustic feature, and vocode them to the voice.

###### Front-end and the difference with Hmm based model:

Since Merlin is not a complete TTS system, it only provides the core acoustic modeling module of TTS (acoustic and speech feature normalization, neural network acoustic model training and generation). So the front-end of it just uses the Festival, and you can also find that in the process.

So the task of describing front-end of merlin is basically turn to the task of describing Festival. And we have learnt it during the speech synthesis 1 as in the vocoder topic.

Festival is a front-end modules. It have the module: Text, which do the whitespace tokenization, Token\_POS, which do the basic token identification, Token, for verbalization, POS, speech tagger, Phrasify, for predicting the phrase breaks, Word, Pauses, which predict the pause, PostLex, modifying segments based on context.

And the difference of Front-end in Merlin and HMM based model is actually the difference between the DNN model and HMM model. For HMM model, you need to do the regression tree to decide which HMM model it gonna use, and for DNN model, you just map the linguistic features to the acoustic features. And another difference I assume, is that in the Merlin model need to add the duration information to the txt information for training the duration and acoustic model.

##### Reflection on the output

Instead of calling them arctic voice, I would prefer call them robotic voice, since it sounds much like the no feeling robot, and I think for synthesize robotic voice, merlin did a great job.

And there is a clear difference in voice quality between the vowel and consonant, it seems to me for the vowels, even the diphthong, merlin did a great job, sounds clear enough for me. But for the consonant, the friction in the consonant doesn’t very clear, especially for the double consonants, I, as not a native speaker, find it is hard to recognize consonant during listening to the wav file.