Contents

[Literature Survey (draft) 1](#_Toc98943512)

[Acoustic phonetics 1](#_Toc98943513)

[Automatic Speech Recognition 1](#_Toc98943514)

[Preprocessing 1](#_Toc98943515)

[Input Format 1](#_Toc98943516)

[Model Architecture 3](#_Toc98943517)

[Language Model 5](#_Toc98943518)

[Performance Metrics 5](#_Toc98943519)

[Text to Speech 6](#_Toc98943520)

[Overview of Speech System 7](#_Toc98943521)

[Capsule Neural Network 8](#_Toc98943522)

[Methodology (plan) 9](#_Toc98943523)

[ASR only 9](#_Toc98943524)

[ASR+TTS 10](#_Toc98943525)

# Literature Survey (draft)

## Acoustic phonetics

## Automatic Speech Recognition

Automatic Speech Recognition (ASR) is a technology that uses machines to convert human speech into text, which can be traced back to the Audrey speech recognition system developed by Bell Labs in the 1950s.

Dynamic time warping (DWT) was the mainstream of ASR before Hidden Markov model (HMM) was introduced into ASR in the 1980s and replaced it. Then neural network was introduced and used widely. In 2010, the model of combining DNN and HMM achieved success, and some recent ASR works began to focus on end-to-end models.

Modern ASR models normally have five parts to consider: preprocessing, input format, model architecture, language model and performance metrics, as shown in Figure 2.1.

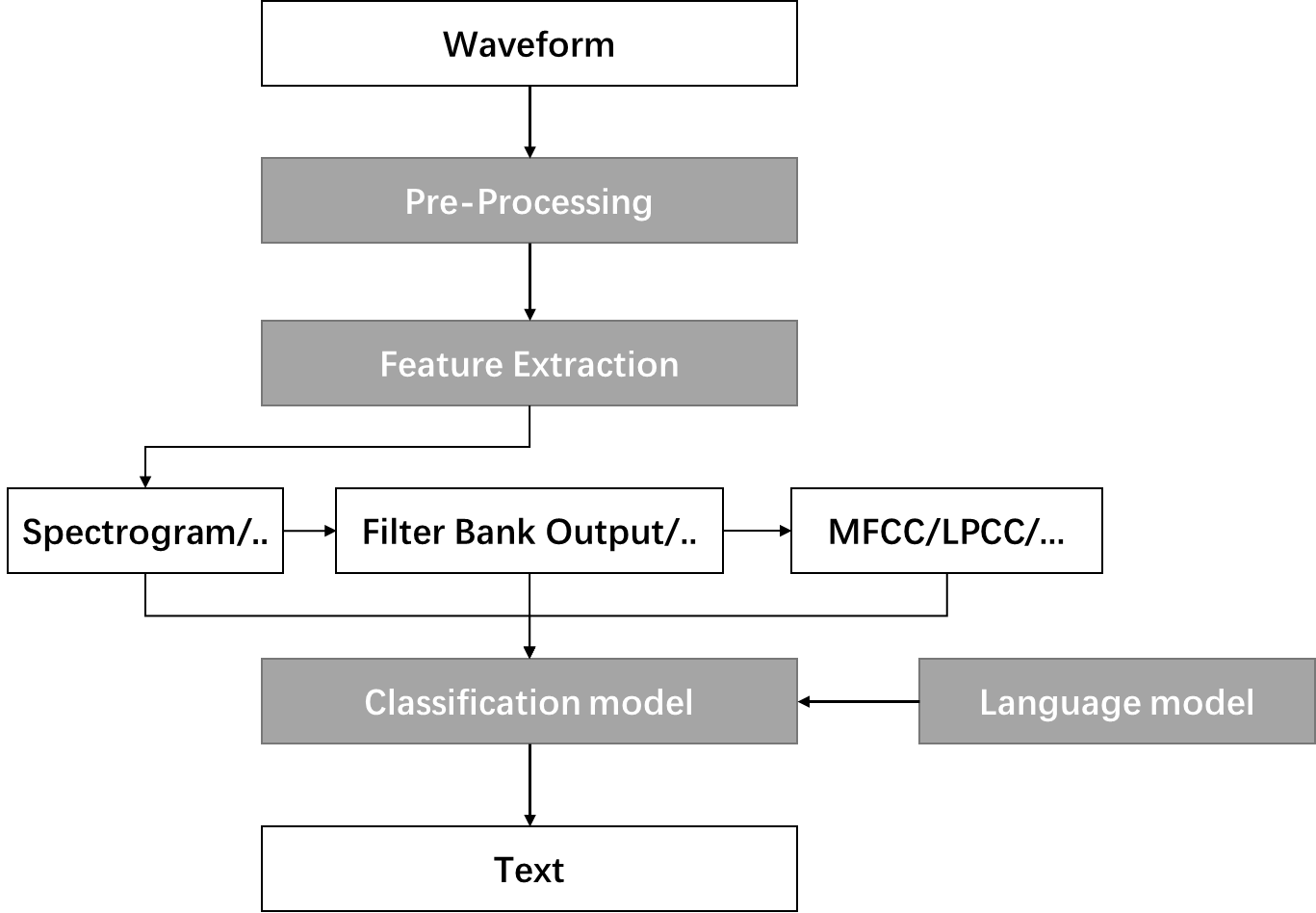


Figure 2.1: The structure of basic ASR model.

### Preprocessing

The preprocessing methods are those methods that have no fundamental relation of the input format, including framing, normalization, end-point detection and pre-emphasis.

### Input Format

Input format, also called acoustic feature, represent the final input of the model architecture. It also distinguishes whether the model is end-to-end. There are four typical formats including waveform, spectrogram, filter bank output and mel-frequency cepstral coefficients (MFCC).

#### Waveform

Waveform is the graphical representation of the shape and form of the sound signal, which means it is the original form of the raw data without any transformation.

#### Spectrogram

Spectrogram is a visual representation of the spectrum of frequencies of a signal as it varies with time. It is normally generated by Fourier transform.

##### Formant

A formant, a dark band on a wide band spectrogram, is the broad spectral maximum that results from an acoustic resonance of the human vocal tract.

The corresponding formants on the spectrogram of different voices are also different, so formant is a very useful feature in ASR and TTS.

A picture containing background pattern

Description automatically generated

Figure 2.2: Various kinds of sounds appear on a spectrogram. F1 and F2 are two formants with different frequency.

#### Filter Bank Output

A filter bank is an array of bandpass filters that separates the input signal into multiple components, each one carrying a single frequency sub-band of the original signal. Filter bank output is widely used in ASR model.

#### Mel-frequency Cepstral Coefficients

After applying discrete cosine transform (DCT) to decorrelate the filter bank output, we get MFCC as another form of input, which used to be popular in ASR. However, due to the pursue of end-to-end models, MFCC is already taken place by previous three input formats.

### Model Architecture

Earlier ASR models used HMM and ANN as the main architecture of the model, and later RNN and LSTM were introduced into ASR making the end-to-end model gradually became the mainstream. This paper mainly analyzes several classic end-to-end model structures, especially those that use transformers as encoder/decoder.

#### Connectionist Temporal Classification (CTC)

CTC is an output model proposed as early as 2006 to solve the problem of correspondence and repetition in sequence prediction. The classic CTC model uses the Mel filter bank output as the input of the network, uses the RNN as the encoder, and finally uses the CTC to obtain the prediction result.

The idea of CTC is to add a Ɛ as a marker to the result of the predicted sequence. Ɛ itself has no meaning, but all the same output between two Ɛ is regarded as the same result and only one text is output.

#### Listen, Attend and Spell (LAS)

LAS is an earlier end-to-end ASR model that uses an attention-based encoder-decoder model. The encoder (Listen) part uses down-sampling such as pyramid RNN and pooling over time to shorten the sequence length. The encoded result is input to the RNN decoder (Spell) through the dot-product attention module.

In the model training design, beam search and teacher forcing are used to ensure that the model can get the expected results. Beam search is to solve the problem that the global optimal solution may not be obtained when the model sequence outputs the results, and the solution it provides is to retain the top N solutions when selecting the results to ensure the stability of the output results. Teacher forcing is designed to solve the situation where the error caused by sequence prediction is amplified eventually making the model untrainable. Since the input of the decoder also includes the output of the previous sequence, once the previous sequence is predicted incorrectly, the subsequent results will be affected. Teacher forcing means that only the truth values of the previous sequence are input into the prediction of the next sequence during training, which ensures that the previous errors will not affect the subsequent results.

#### RNN-T, Transducers and MoChA

When CTC makes predictions, each input will have a prediction result. However, not all inputs correspond to only one output, such as the ‘th’ pronunciation in English. Therefore, RNN-T, Transducers and MoChA are designed to solve this problem. Among them, RNN-T allows one output to generate multiple outputs until the network thinks that the optimal solution is reached; Transducers go a step further on the basis of RNN-T, allowing the model to have multiple inputs at the same time; MoChA adopts a different idea, using Similar to the sliding window mechanism, the model automatically determines when new data is entered.

#### Transformer

Transformer was introduced in 2017 by Google Brain and are increasingly the model of choice for NLP problems, replacing RNN models such as LSTM.

The transformer is also a seq2seq model with an encoder-decoder structure. Its biggest improvement is to replace the previous RNN structure with self-attention. Self-attention has been proven to be a very powerful module, which not only has the same representation ability as RNN and CNN, but also can realize parallel processing of sequence generation. In the previous RNN network, the prediction depends on previous outputs so that the whole sequence is generated sequentially, which is not efficient. The introduction of self-attention solves this problem, which greatly boosts up the training speed.

##### Self-Attention

Self-attention is a technique for processing sequence data and obtaining the relationship between data, which is very effective in processing seq2seq tasks. Although the processing of self-attention does not consider the order of the sequence, it can still be used well for ASR with the additional input of position information.

The key components of self-attention are query, key and value vectors, which are also the parameters that need to be trained in the self-attention model. For each segment in the sequence, there are three vectors Q, K and V. The output is to calculate the correlation between Q and K corresponding to other data of the sequence, and finally use the correlation to weight V to obtain the output.

图示

描述已自动生成

Figure 2.3: The demonstration of self-attention by Voita (https://lena-voita.github.io/nlp\_course.html)

##### Structure

Both encoder and decoder of transformer are built by one or two self-attention blocks. An encoder that reads the text input and a decoder that produces a prediction for the task.

图示

描述已自动生成

Figure 2.4: The structure of speech transformer by Li, Jie et al. (The Speech transformer for Large-scale Mandarin Chinese Speech Recognition)

### Language Model

Language model is a probability distribution over sequences of words. It obtains more probable semantic features by analyzing large amount of text data to correct the prediction of the model.

Before the widespread use of neural networks, the more commonly used language model is n-grams. Continuous space language models are often used in current neural network models.

### Performance Metrics

The last part of ASR is the metrics of model performance. It mainly includes speed and accuracy. Speed metrics indicate the prediction speed of the model. Commonly used metrics are text entry rate in words per minute (wpm) and real-time factor (RTF).

## Text to Speech

图示

描述已自动生成

Figure 2.5: The structure of basic TTS model by Tan, Xu, et al. (A Survey on Neural Speech Synthesis).

## Overview of Speech System

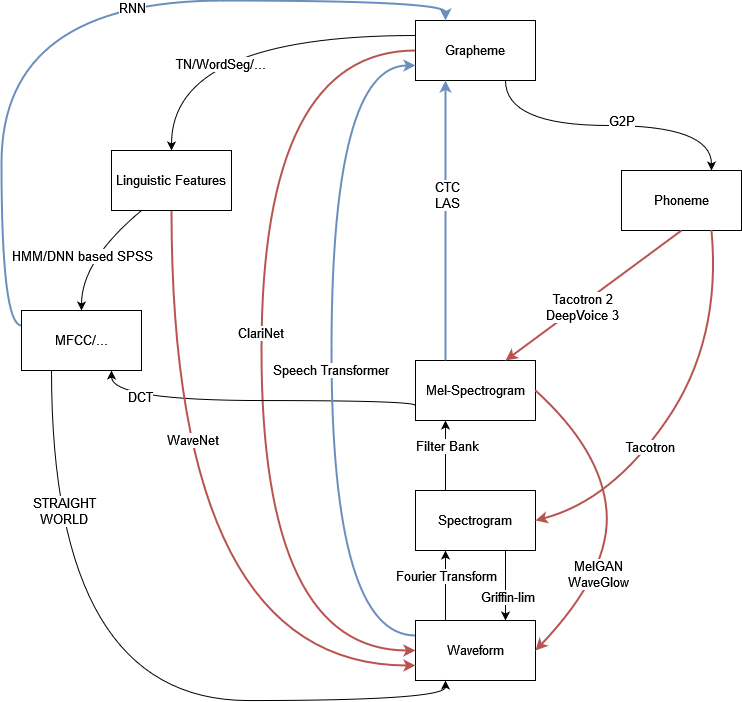


Figure 2.6: Overview of ASR and TTS possible workflow.

## Capsule Neural Network

# Methodology (plan)

## ASR only

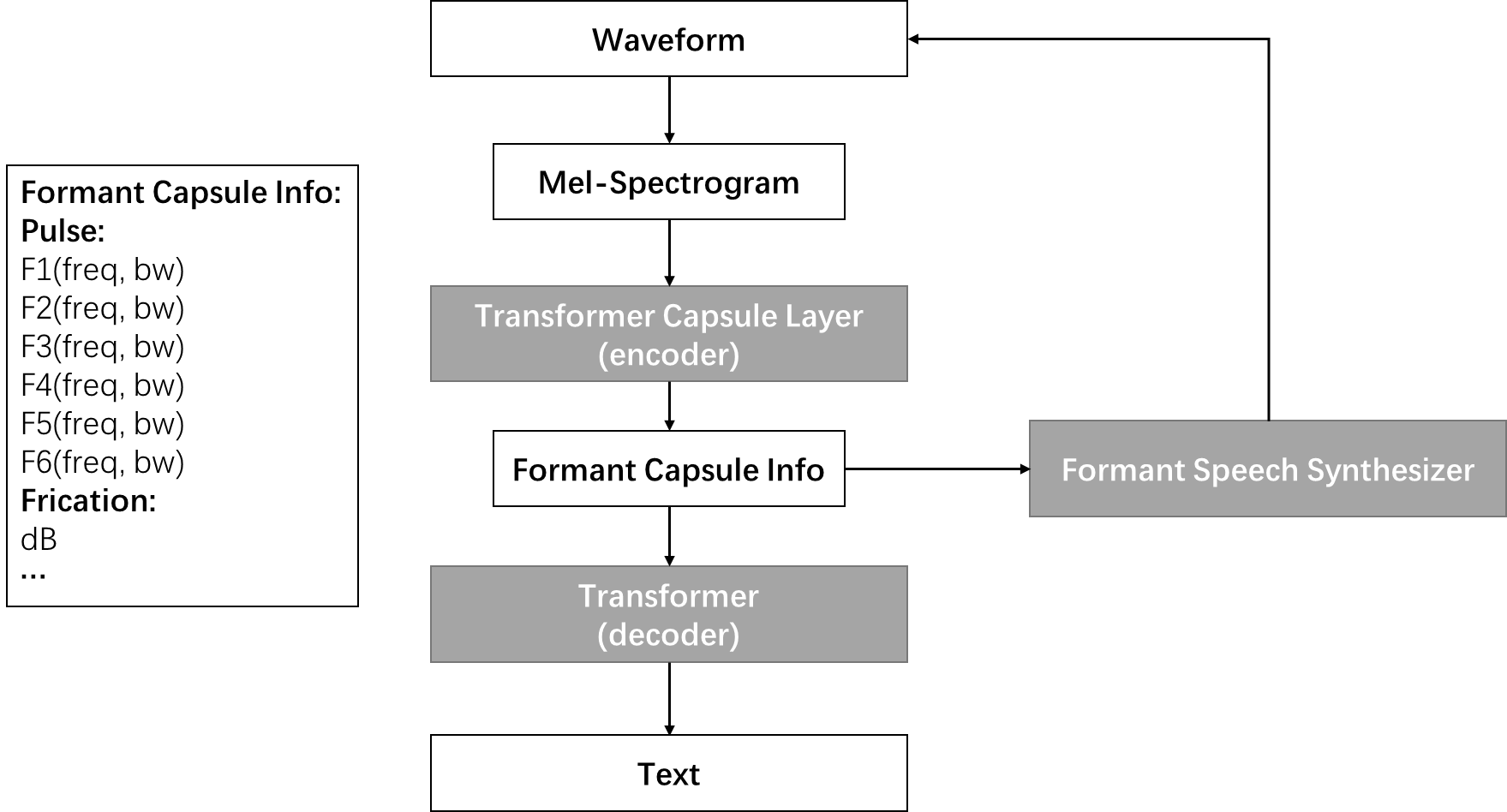


Figure 3.1: Network with only formant capsules. Each capsule contains all the formant information of a short piece of time.

My current thinking is to use a seq2seq network, as shown in Figure 3.1. It has the following process:

* Convert the sound signal into mel-spectrogram
* Input the encoder to get the formant capsule
* Input the formant capsule into the decoder for prediction, or use the formant speech synthesizer to synthesize waveform

The above model cannot solve high level capsule structure like phoneme.

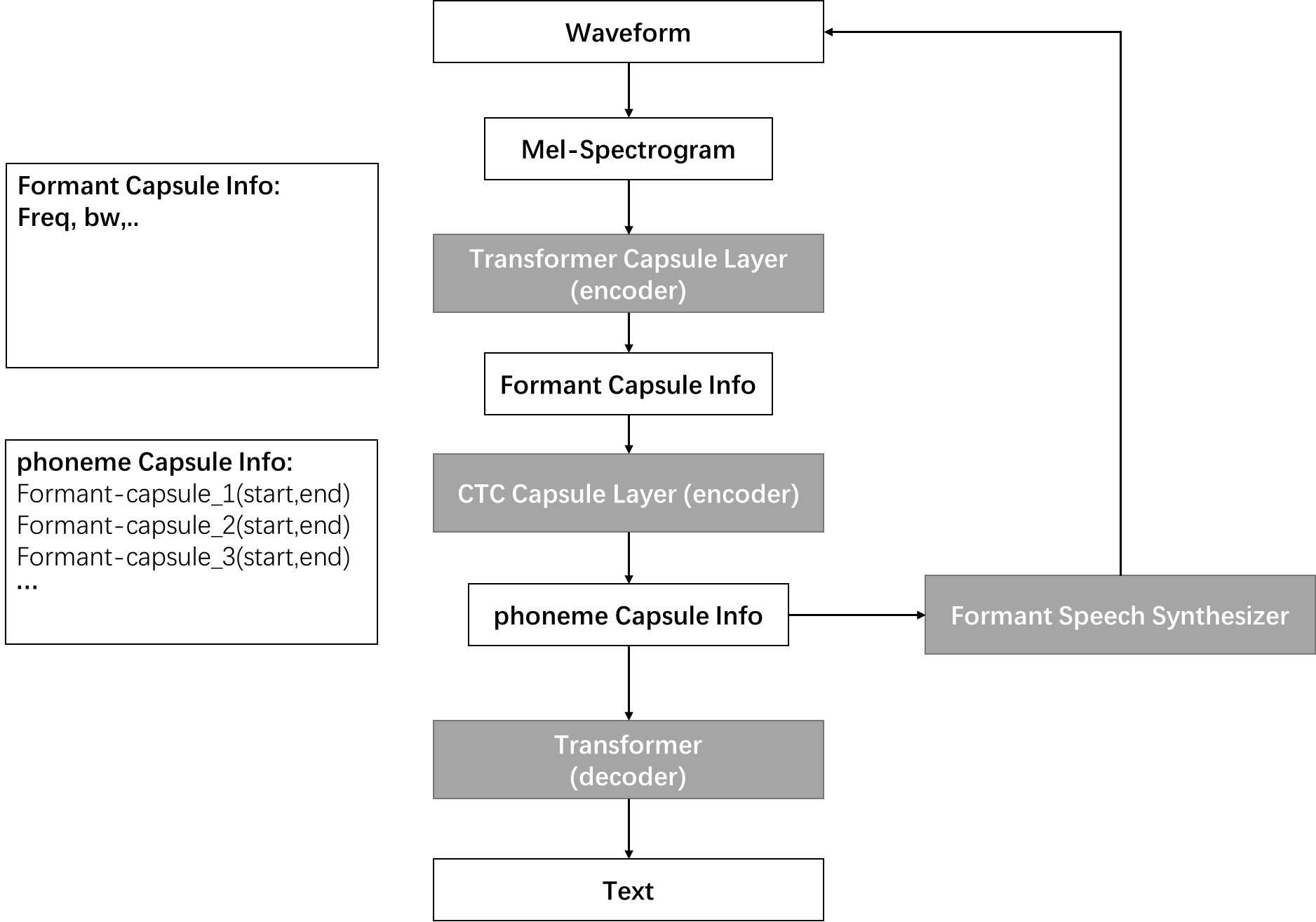


Figure 3.2: Network with two level capsules (formant and phoneme). The formant capsule only contains one formant. The phoneme capsule assembles several formant capsules.

Therefore, in Figure 3.2, I am thinking add another capsule layer to reconstruct the sound of phonemes. At second capsule layer down sample the sequence with CTC-like mechanics. (I doubt the feasibility)

## ASR+TTS

However, the above two models cannot do TTS work. If I try to build both ASR and TTS system, maybe two networks would be needed.

Besides, I am still confused about the correspondence between phoneme and its sound. With capsule network, I only get several capsules without knowing their identities so that there is no way to use them to reconstruct the speech.