**preprocessor**

* **Tokenize**: Tokenize a sentence into words
* **Phonemes/Pronunciation**: It breaks input text into phonemes, based on their pronunciation. For example, “Hello, Have a good day” converts to HH AH0 L OW1, HH AE1 V AH0 G UH1 D D EY1.
* **Phoneme duration:** Represents the total time taken by each phoneme in the audio.
* **Pitch:** Key feature to convey emotions, it greatly affects the speech prosody.
* **Energy:** Indicates frame-level magnitude of mel-spectrograms and directly affects the volume and prosody of speech.

The Linguistic feature only contains phonemes. Energy, pitch, and duration are actually used to train the energy predictor, the pitch predictor, and the duration predictor respectively which are used by the model to get a more natural output.

encoder

The encoder inputs Linguistic features (Phonemes) and outputs an n-dimensional embedding. This embedding between the encoder and decoder is known as the latent feature. Latent features are crucial because, other features like speaker embedding (will be explained in the coming blog) are concatenated with these and passed to the decoder. Furthermore, the latent features are also used for the prediction of energy, pitch, and duration which in turn play a crucial role in controlling the naturalness of the audio.

decoder

The decoder is used to convert information embedded in the Latent processed feature to the Acoustic feature i.e. Mel-spectrogram.

But why output mel-spectrograms instead of directly producing the speech/audio from the decoder?

This is because audio contains more variance information (e.g., phase) than Mel-spectrograms. This causes a larger information gap between the input and output for text-to-audio as compared to text-to-spectrogram generation. Hence, Mel-spectrograms are preferred.

vocoder

It converts the Acoustic feature (Mel-spectrogram) to waveform output (audio). It can be done using a mathematical model like Griffin Lim or we can also train a neural network to learn the mapping from mel-spectrogram to waveforms. In reality, learning-based methods usually outperform the Griffin Lim method.

So instead of directly predicting waveform using the decoder, we split this complex and sophisticated task into two stages, first predicting mel-spectrogram from Latent processed features and then generating audio using mel-spectrogram.