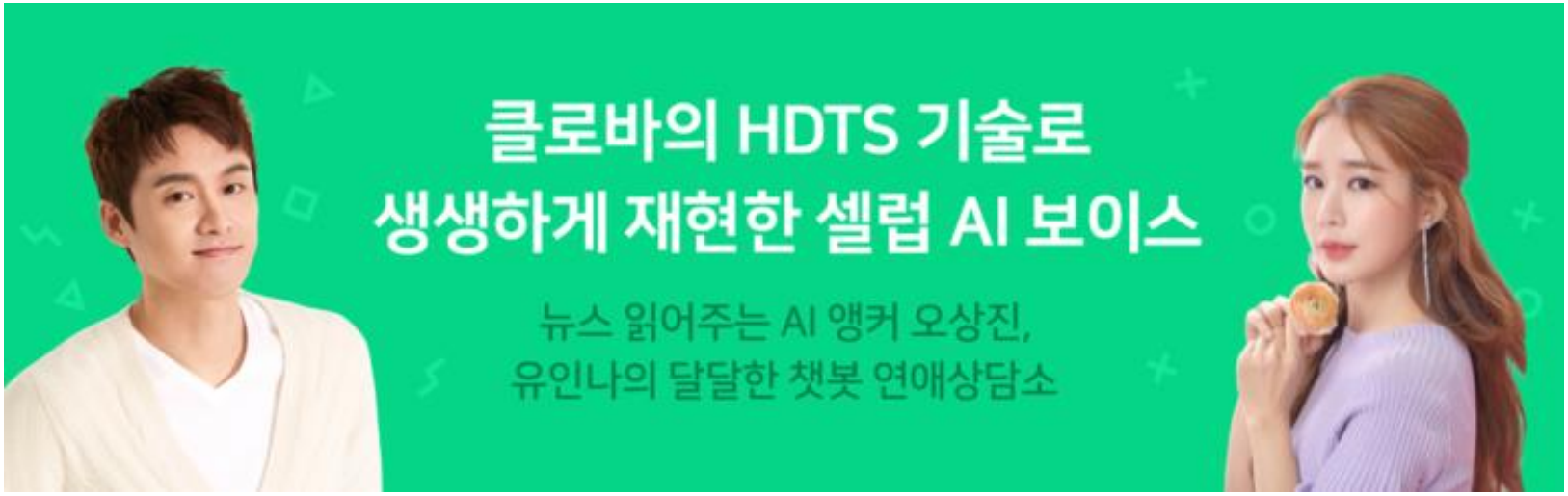
CLOVA

딥러닝 기반 음성 합성 시스템

송은우 / NAVER CLOVA

Introduction

Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.



Introduction

Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.

DNN TTS = Acoustic model + Vocoder

Introduction

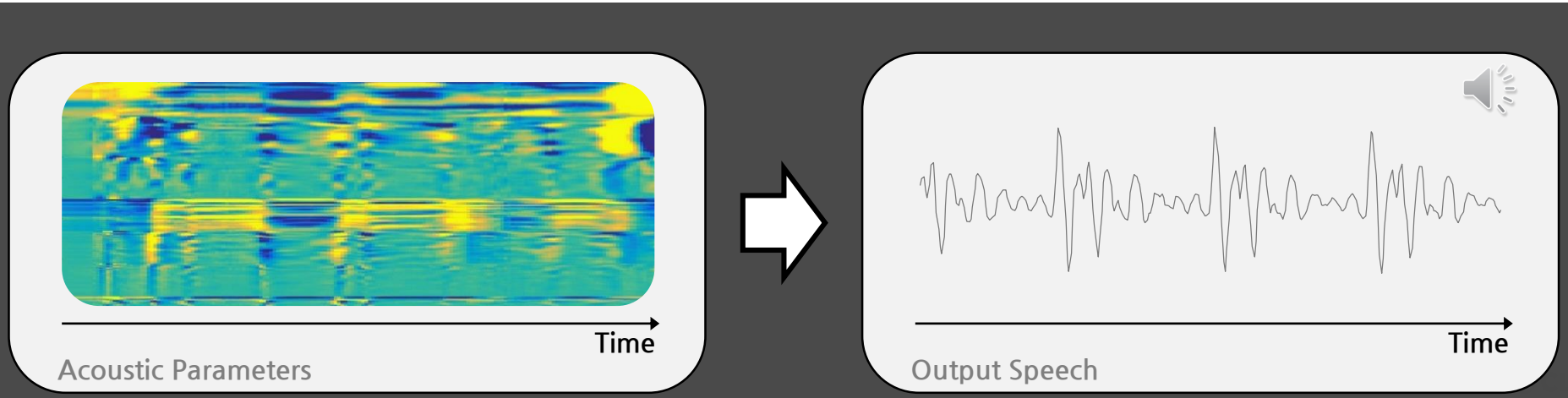
Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.

톤의 높낮이, 음색, 어조, 강세 등
텍스트에서 **Acoustic Parameter** 를 추정

Introduction

Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.

Acoustic Parameter 에서 음성 신호를 생성



Introduction

Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.

Acoustic Parameter 에서 음성 신호를 추정

본 발표에서는 TTS 엔진의 핵심 요소인

Acoustic Model & Vocoder 기술을 정리하고,

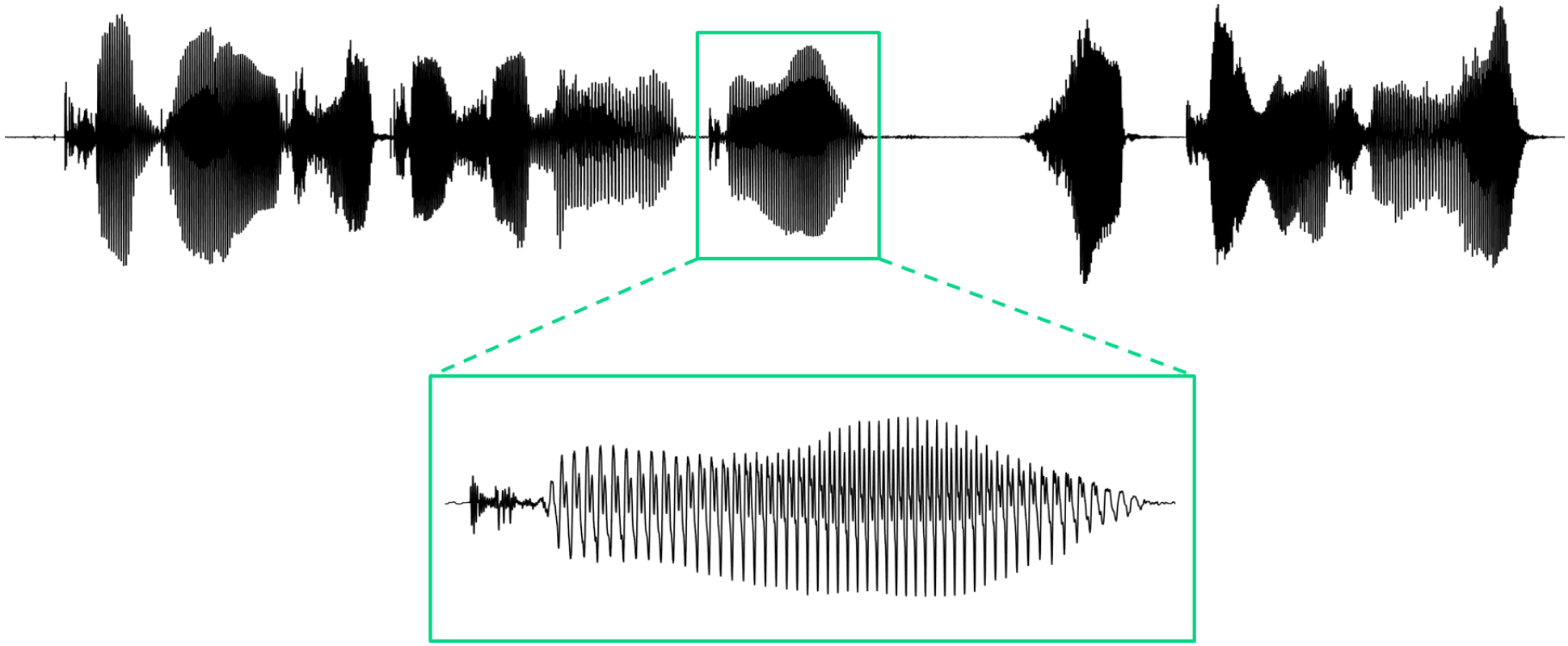
CLOVA 의 다양한 음성 합성 서비스를 소개하고자 합니다.

Speech fundamentals

What is speech ?

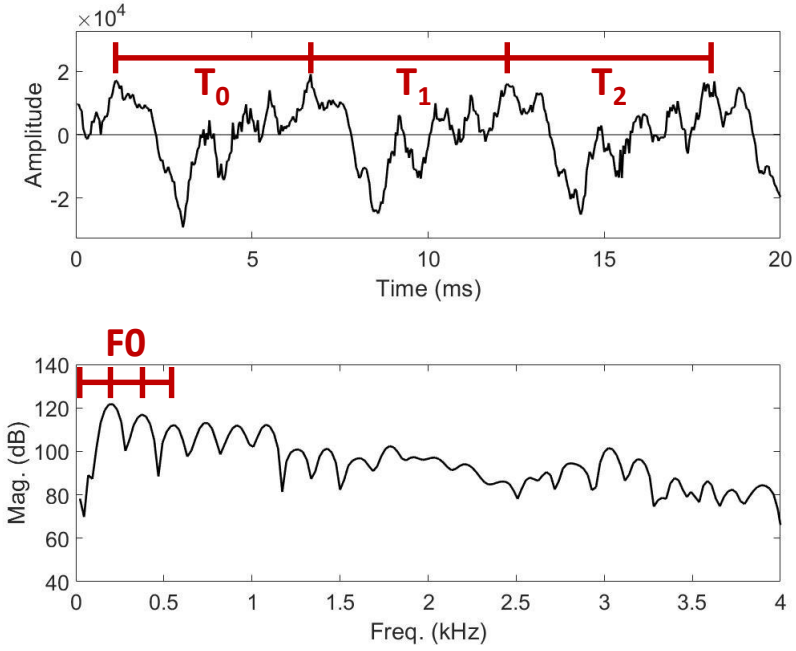
CLOVA

Speech waveform



Pitch period

음성의 **주기성**을 나타내는 파라미터: 음성의 **톤**을 결정합니다 (ex. 하이톤, 중저음).



Pitch period = $T\_0 \approx T\_1 \approx T\_2$

- Long-term period of speech (time-domain)

Fundamental frequency (F0) = $1/T\_0$

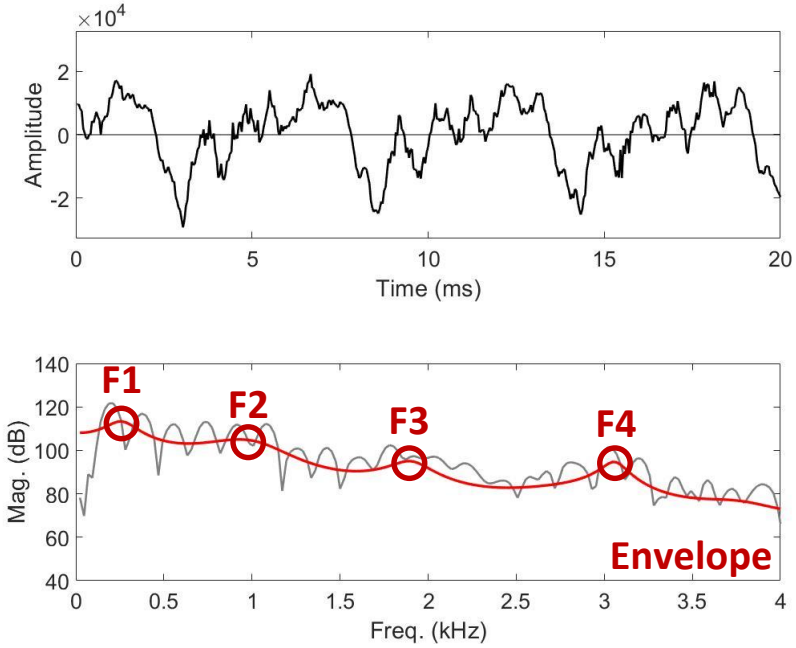
- 1 / PP (frequency-domain)
- Female voice: Ave. 200 Hz
- Male voice : Ave. 100 Hz

Harmonic spectrum

- Multiple peaks of speech spectrum (interval=F0)

Formant frequency

음색을 나타내는 파라미터: 음성의 **발음**을 결정합니다 (ex. 아 / 에 / 이 / 오 / 우).



Pitch period = $T\_0 \approx T\_1 \approx T\_2$

- Long-term period of speech (time-domain)

Fundamental frequency (F0) = $1/T\_0$

- 1 / PP (frequency-domain)
- Female voice: Ave. 200 Hz
- Male voice : Ave. 100 Hz

Harmonic spectrum

- Multiple peaks of speech spectrum (interval=F0)

Formant frequency (F1, F2, ...)

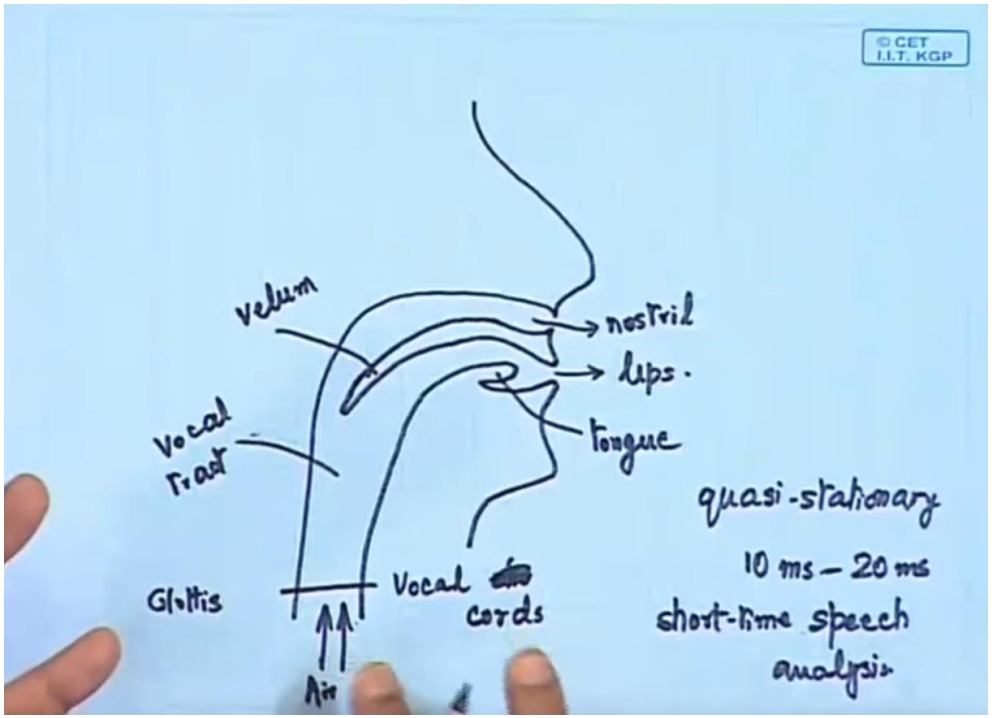
- Vocal tract resonance

Speech fundamentals

How do we produce speech ?

How do we produce speech?

Speech Production Model

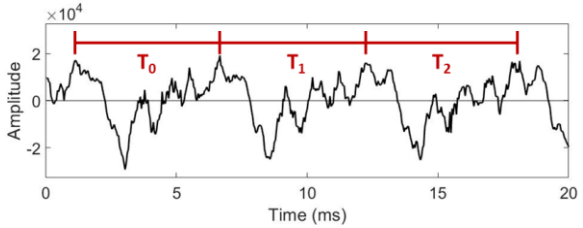


https://www.youtube.com/watch?v=X_JvfZiGEek

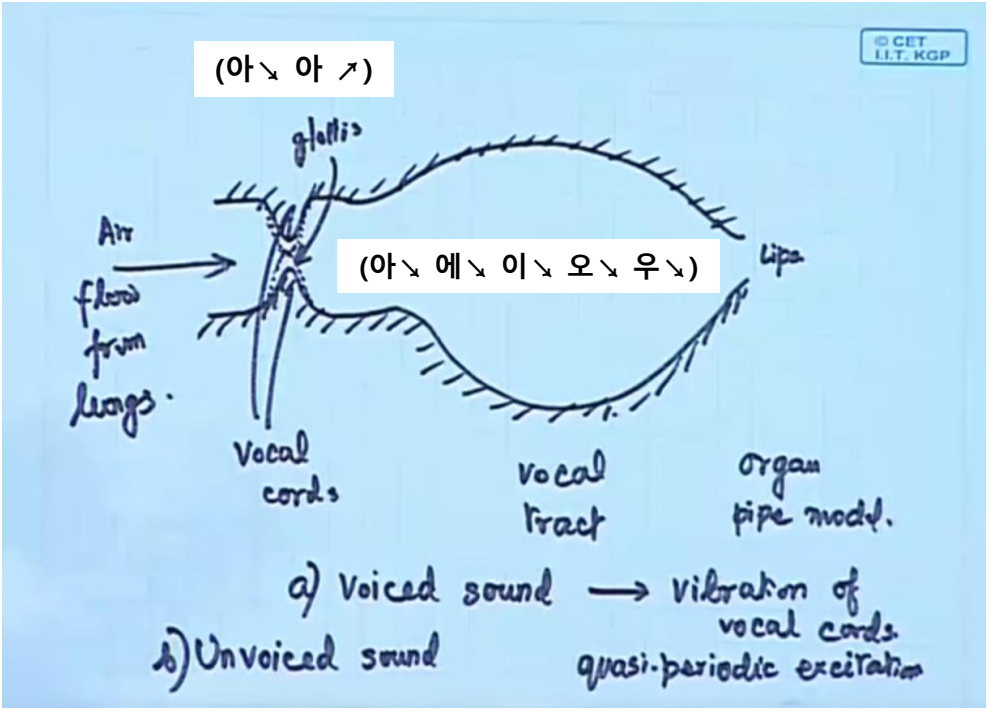
Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

How do we produce speech?



Speech Production Model



https://www.youtube.com/watch?v=X_JvfZiGEek

Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

How do we produce speech?

Speech Production Model: Linear Prediction

Linear prediction

- Representation of speech
  - Weighted sum. of previous samples.
    - $\hat{s}(n) = \sum_{k=1}^{p} a(k)s(n-k)$
- Prediction error
  - Time-domain
    - $e(n) = s(n) - \hat{s}(n) = s(n) - \sum_{k=1}^{p} a(k)s(n-k)$
  - Minimizing mean square error
    - $\underset{a_k}{\operatorname{argmin}}\, E\left\{\left\|s(n) - \sum_{k=1}^{p} a(k)s(n-k)\right\|^2\right\}$

Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

How do we produce speech?

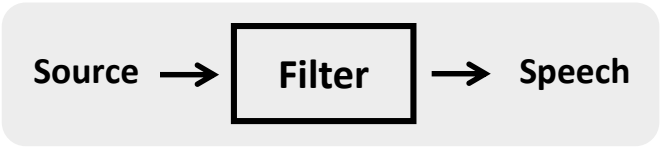
Speech Production Model: Linear Prediction

Linear prediction

- Representation of speech
  - Weighted sum. of previous samples.
    - $\hat{s}(n) = \sum_{k=1}^{p} a(k)s(n-k)$
- Prediction error
  - Frequency-domain
    - $E(z) = S(z) - \sum_{k=1}^{p} a(k)z^{-k}S(z)$
      $= S(z)\left(1 - \sum_{k=1}^{p} a_k z^{-k}\right)$
    - $S(z) = \frac{E(z)}{1-\sum_{k=1}^{p} a_k z^{-k}} = \frac{E(z)}{A(z)} = E(z)H(z)$
    - $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

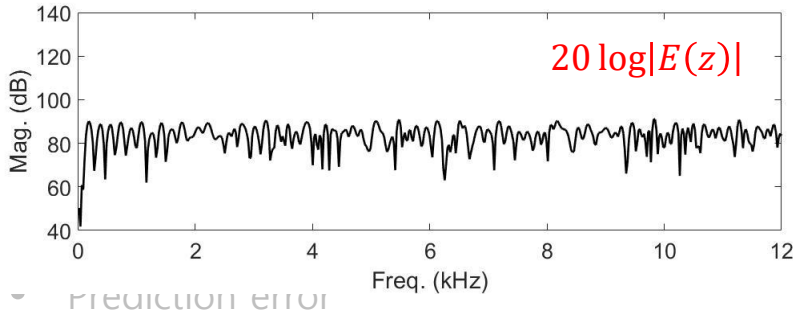
Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter



How do we produce speech?

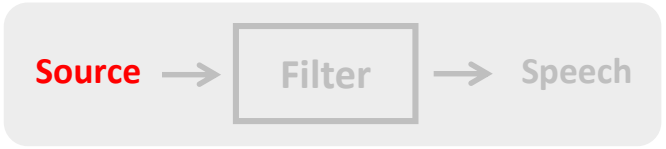
Speech Production Model: Linear Prediction



- Prediction error
  - Frequency-domain
    - $E(z) = S(z) - \sum_{k=1}^{p} a(k) z^{-k} S(z)$
      $= S(z)\left(1 - \sum_{k=1}^{p} a_k z^{-k}\right)$
    - $S(z) = \frac{E(z)}{1-\sum_{k=1}^{p} a_k z^{-k}} = \frac{E(z)}{A(z)} = E(z)H(z)$
    - $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

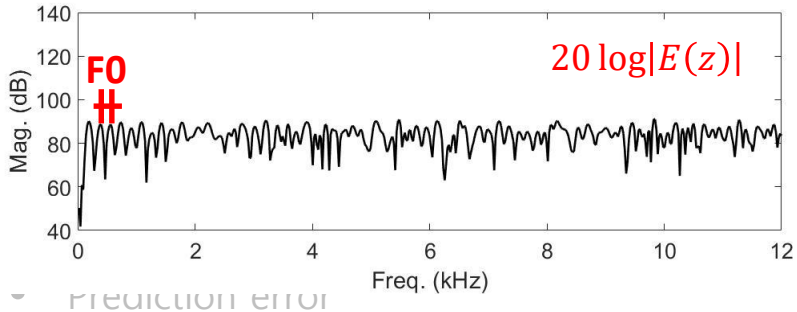
Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter



How do we produce speech?

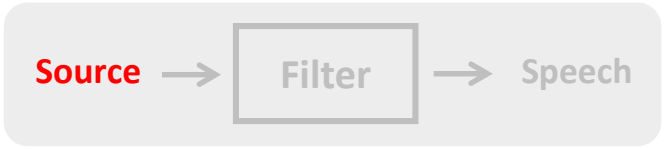
Speech Production Model: Linear Prediction



- Prediction error
  - Frequency-domain
    - $E(z) = S(z) - \sum_{k=1}^{p} a(k) z^{-k} S(z)$
      $= S(z)\left(1 - \sum_{k=1}^{p} a_k z^{-k}\right)$
    - $S(z) = \frac{E(z)}{1-\sum_{k=1}^{p} a_k z^{-k}} = \frac{E(z)}{A(z)} = E(z)H(z)$
    - $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

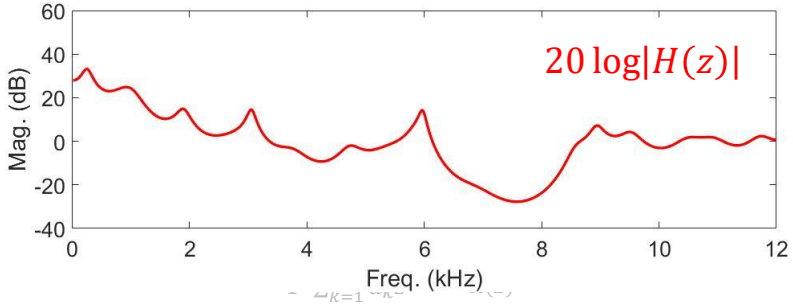


How do we produce speech?

Speech Production Model: Linear Prediction

Linear prediction

- Representation of speech
  - Weighted sum. of previous samples.
    - $\hat{s}(n) = \sum_{k=1}^{p} a(k)s(n-k)$
- Prediction error



  - $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

Source-filter model

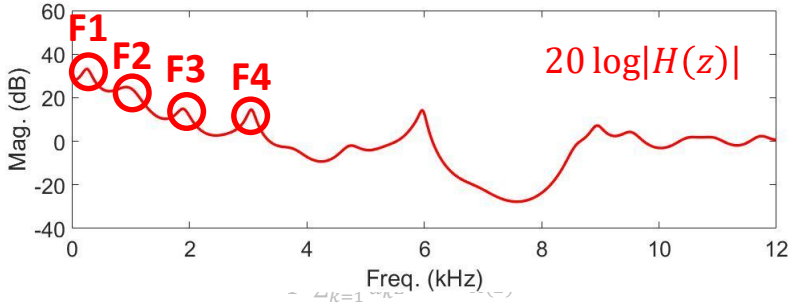
- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

How do we produce speech?

Speech Production Model: Linear Prediction

Linear prediction

- Representation of speech
  - Weighted sum. of previous samples.
    - $\hat{s}(n) = \sum_{k=1}^{p} a(k)s(n-k)$
- Prediction error



  - $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

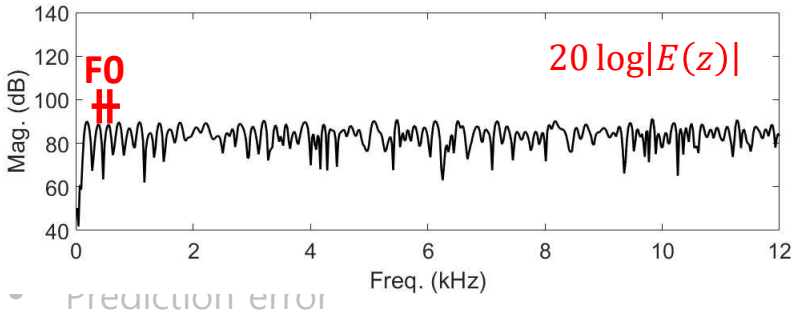
Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

Source → **Filter** → Speech

How do we produce speech?

Speech Production Model: Linear Prediction



- $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

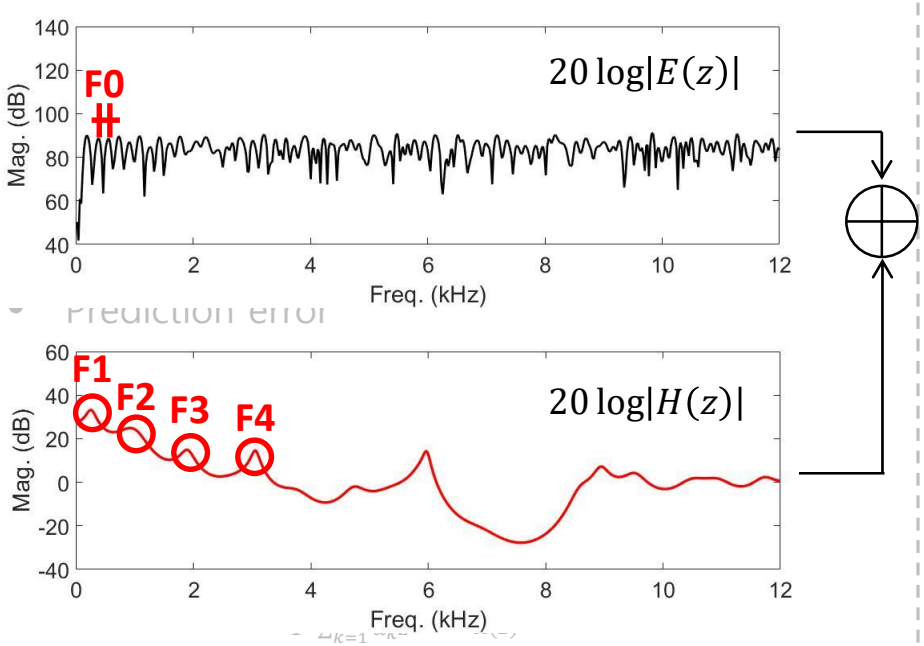
Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

Source → **Filter** → Speech

How do we produce speech?

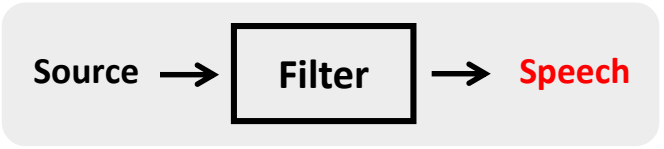
Speech Production Model: Linear Prediction



- $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

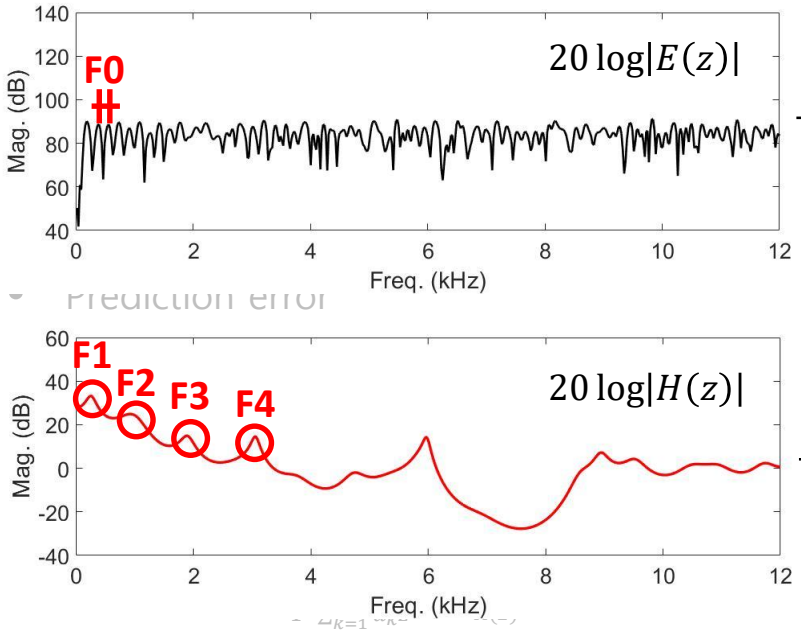
Source-filter model

- Lung
  - Power supply
- Glottis ≈ vocal cords ≈ vocal folds
  - Modulator (= source = excitation)
  - Voiced sound : quasi-periodic
  - Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter



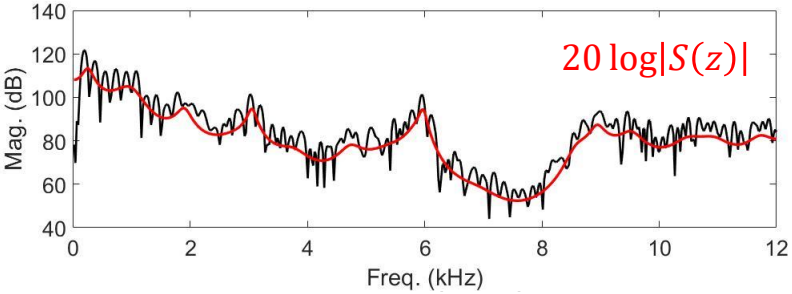
How do we produce speech?

Speech Production Model: Linear Prediction



- $20\log|S(z)| = 20\log|E(z)| + 20\log|H(z)|$

Source-filter model

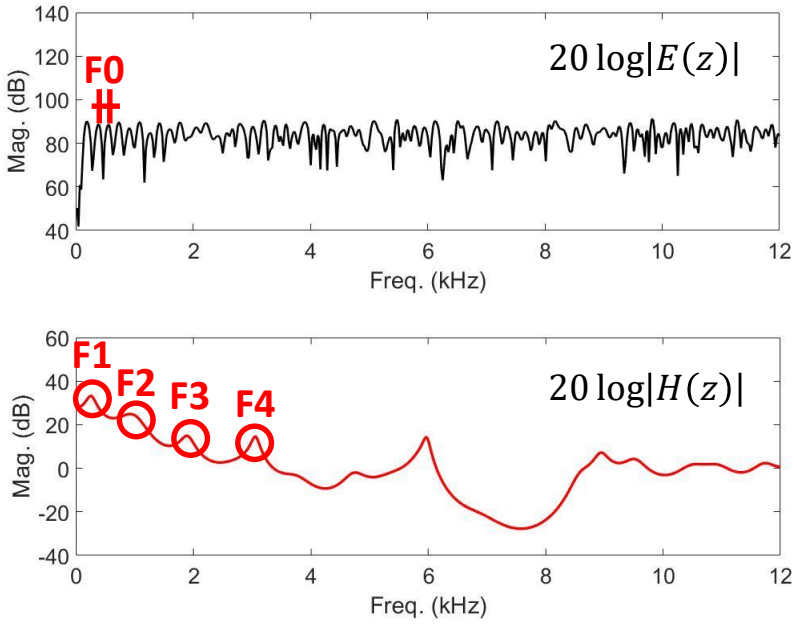


- Unvoiced sound : noisy
- Vocal tract (from vocal folds to lips)
  - Filter

Source → **Filter** → Speech

Summary

Pitch Period (or F0) 와 Linear Prediction 을 꼭 기억해 주세요!



Pitch period

- Long-term period of speech (time-domain)

Fundamental frequency (F0)

- 1 / PP (frequency-domain)

Harmonic spectrum

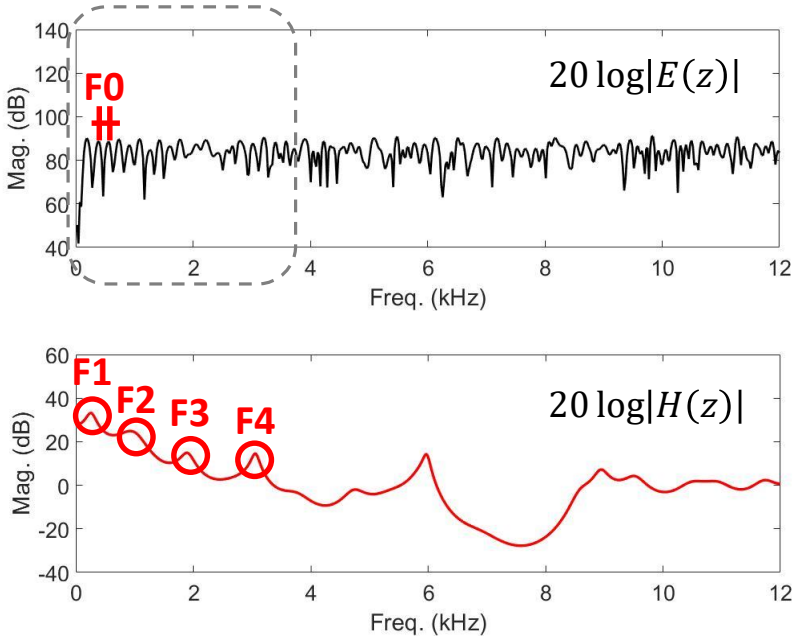
- Multiple peaks of speech spectrum (interval=F0)

Formant frequency (F1, F2, ...)

- Vocal tract resonance

Summary

Pitch Period (or F0) 와 Linear Prediction 을 꼭 기억해 주세요!



Source-filter model

- Glottis ≈ vocal cords ≈ vocal folds
  - Excitation = linear prediction residual

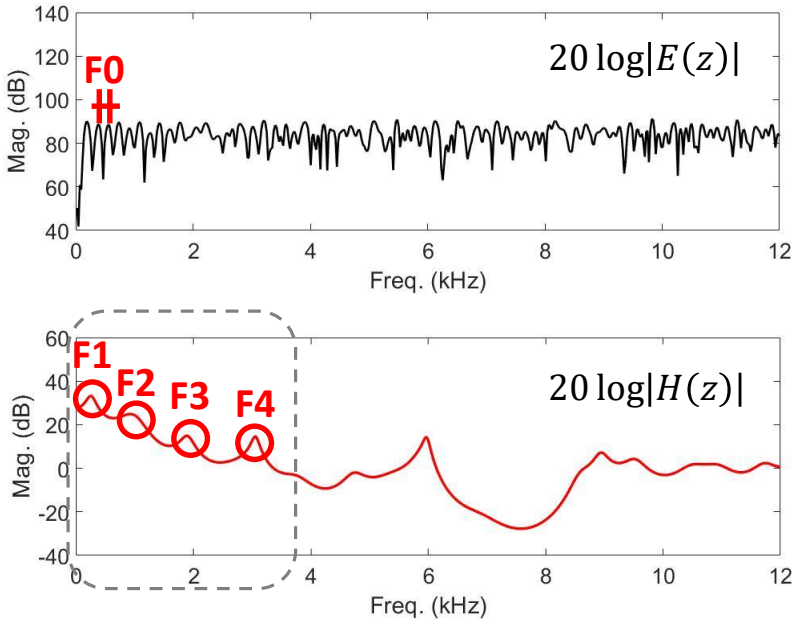
  → Vocal cords movement determines F0
  (아↘ 아↗)

- Vocal tract (from vocal folds to lips)
  - Linear prediction filter

  → LP spectrum determines fomant structure
  (아↘ 에↘ 이↘ 오↘ 우↘)

Summary

Pitch Period (or F0) 와 Linear Prediction 을 꼭 기억해 주세요!



Source-filter model

- Glottis ≈ vocal cords ≈ vocal folds
  - Excitation = linear prediction residual

  → Vocal cords movement determines F0
  (아↘ 아↗)

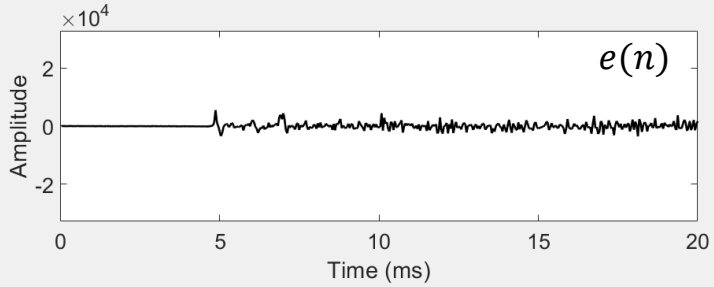
- Vocal tract (from vocal folds to lips)
  - Linear prediction filter

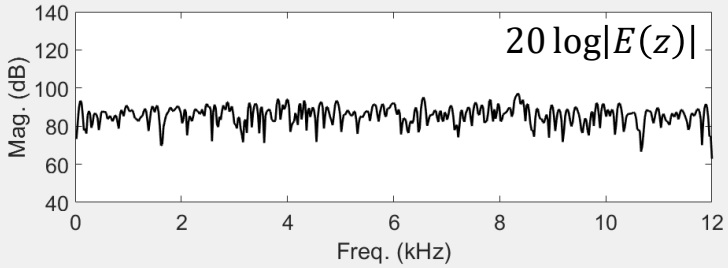
  → LP spectrum determines fomant structure
  (아↘ 에↘ 이↘ 오↘ 우↘)

Summary

Time-frequency analysis of speech production model

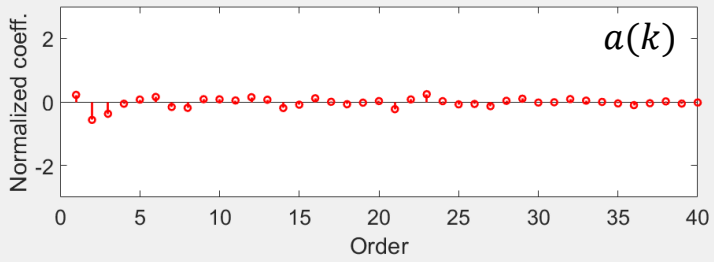
Source
- **Vocal folds**
- **Excitation**
- **LP residual**

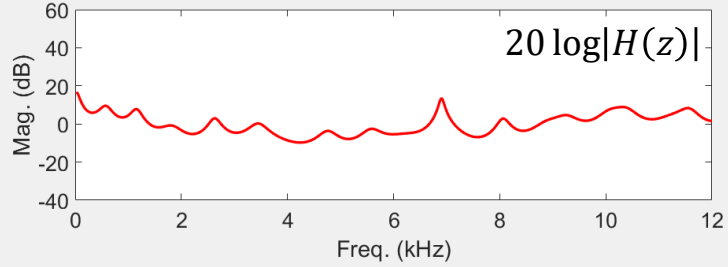




Source
- **Vocal folds**
- **Excitation**
- **LP residual**

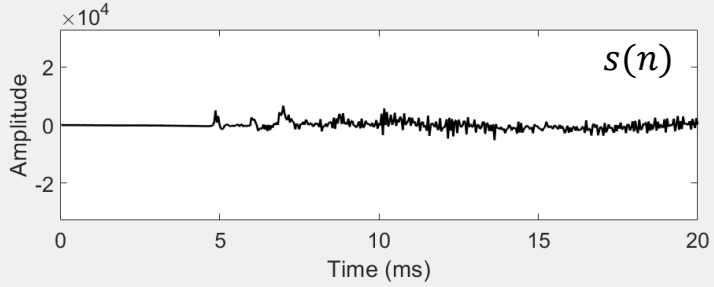
Filter
- **Vocal tract**
- **LP coeff.**

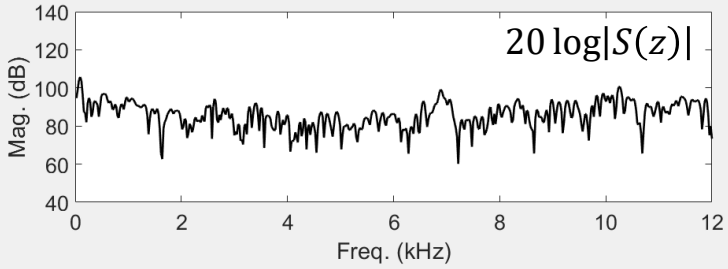




Filter
- **Vocal tract**
- **LP coeff.**

Speech
= e(n) * h(n)





Speech
= E(z)H(z)

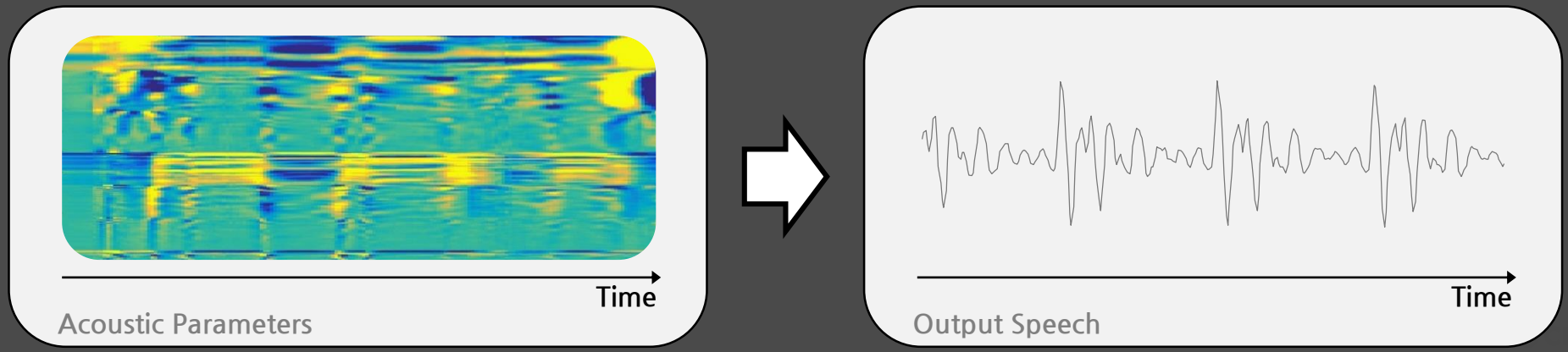
Vocoding model

Parametric LPC vocoder

Recall

Text-to-speech (TTS) 란 기계가 사람처럼 텍스트를 읽어주는 기술입니다.

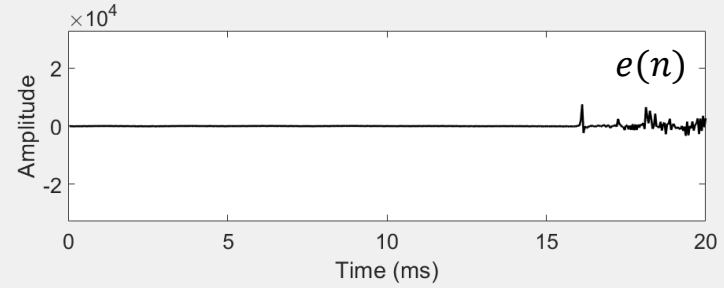
Acoustic Parameter 에서 음성 신호를 생성



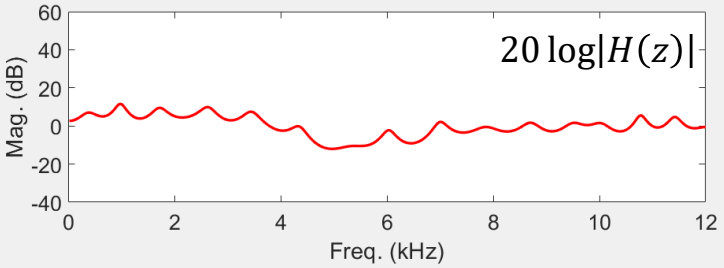
Recall

20 ms 음성 신호를 어떻게 만들 수 있을까요?

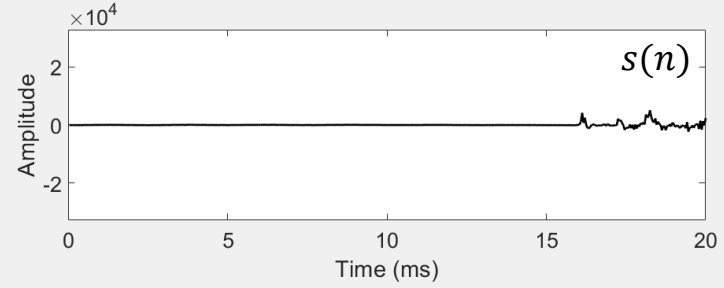
Source
- **Vocal folds**
- **Excitation**
- **LP residual**



Filter
- **Vocal tract**
- **LP coeff.**



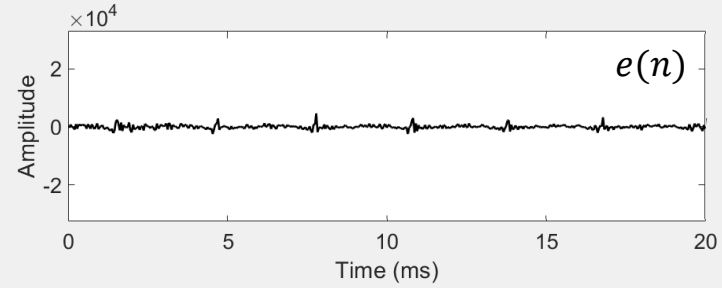
Speech
= e(n) * h(n)



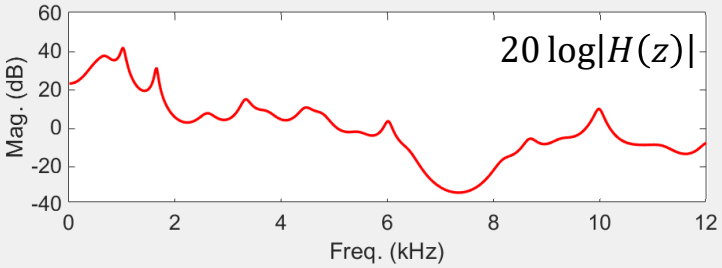
Recall

LP coefficients 40 개

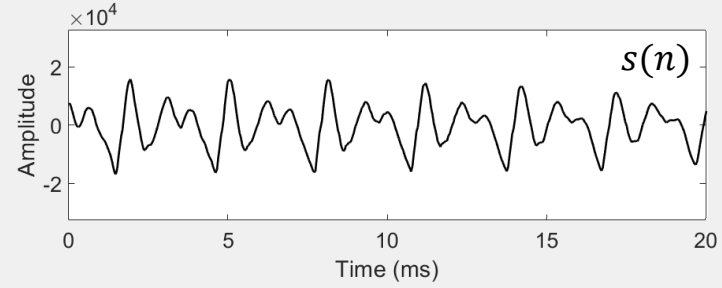
Source
- **Vocal folds**
- **Excitation**
- **LP residual**

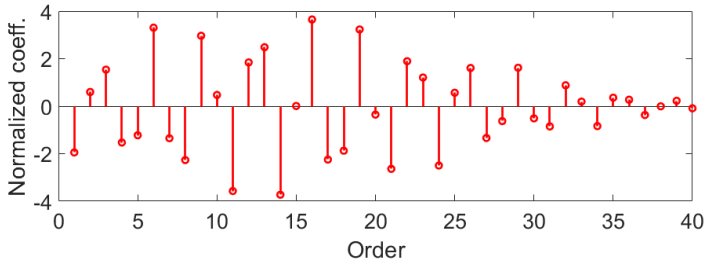


Filter
- **Vocal tract**
- **LP coeff.**



Speech
= e(n) * h(n)



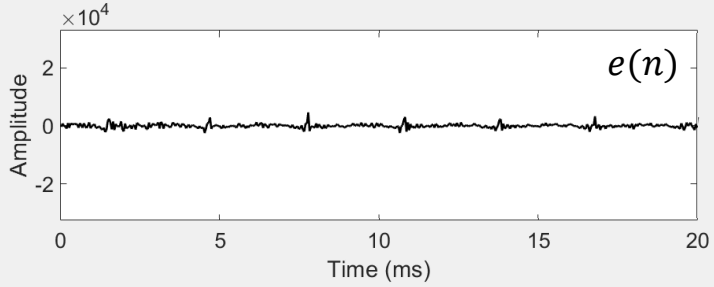


Filter

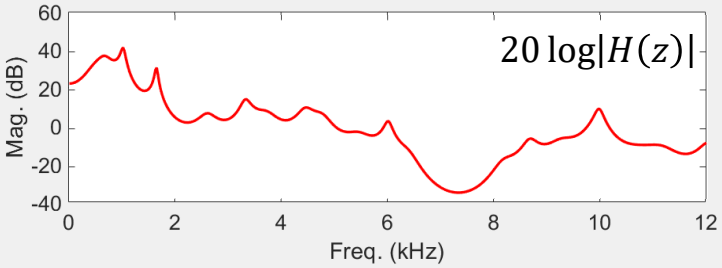
Recall

LP coefficients 40 개 + Excitation 20 ms

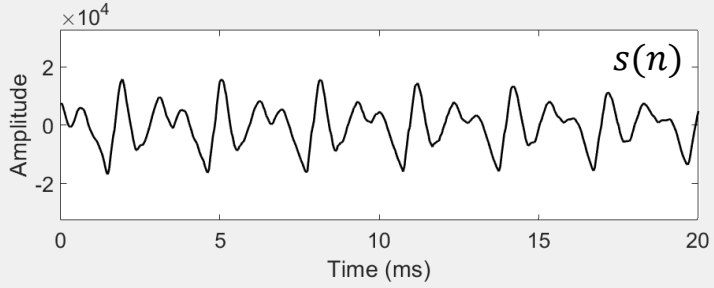
Source
- **Vocal folds**
- **Excitation**
- **LP residual**

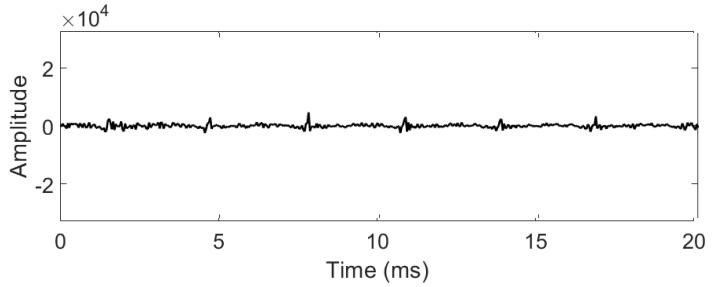


Filter
- **Vocal tract**
- **LP coeff.**

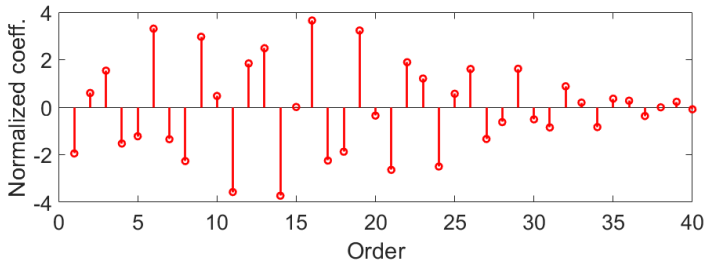


Speech
= e(n) * h(n)





Source

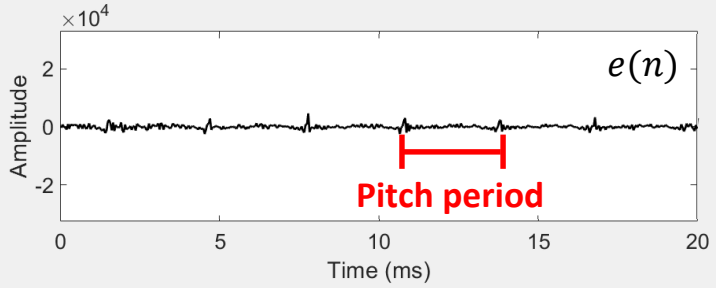


Filter

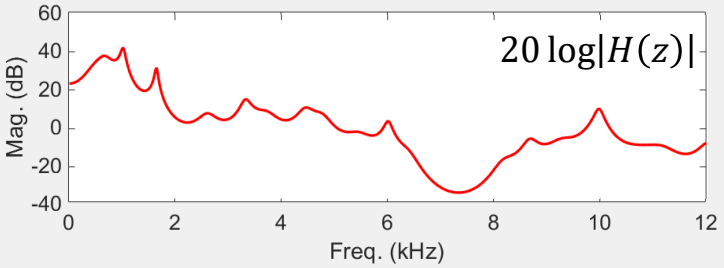
Recall

LP coefficients 40 개 + Excitation 20 ms (approximation using pitch period)

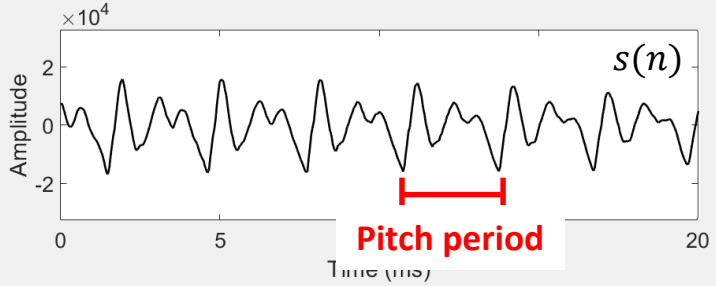
Source
- **Vocal folds**
- **Excitation**
- **LP residual**

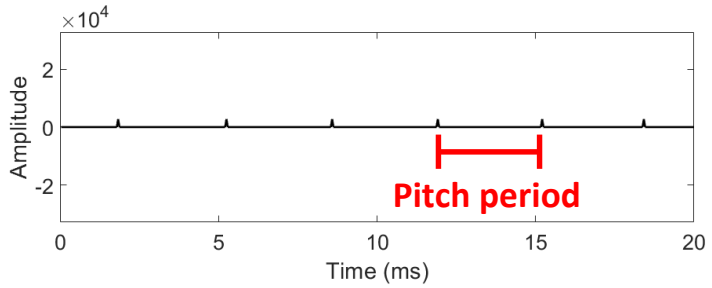


Filter
- **Vocal tract**
- **LP coeff.**

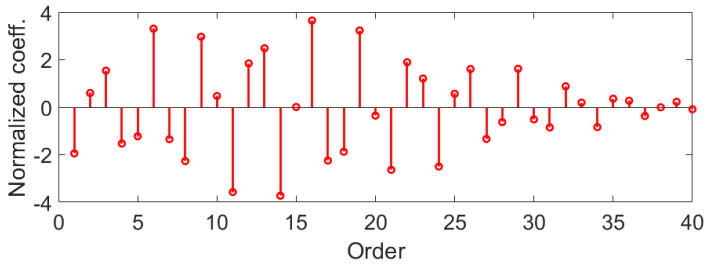


Speech
= e(n) * h(n)





Source

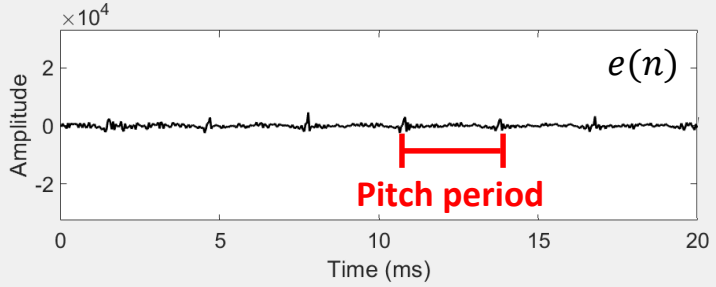


Filter

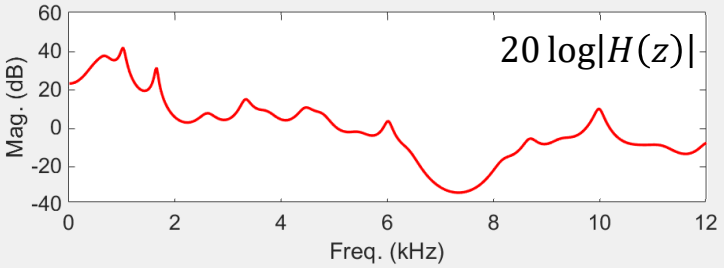
Recall

LP coefficients 40 개 + Excitation 20 ms (approximation using pitch period)

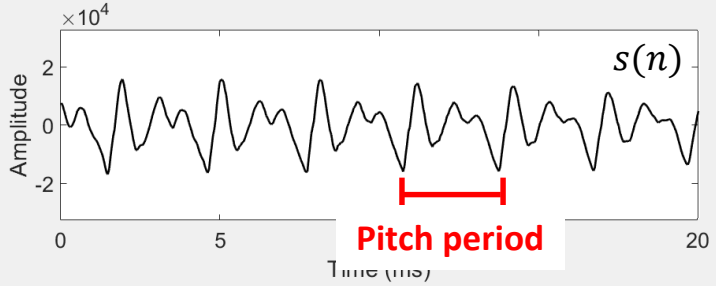
Source
- **Vocal folds**
- **Excitation**
- **LP residual**

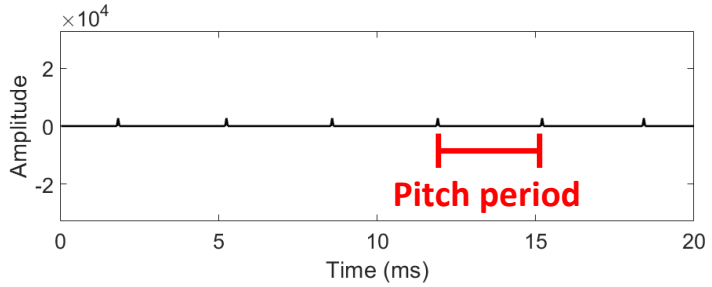


Filter
- **Vocal tract**
- **LP coeff.**

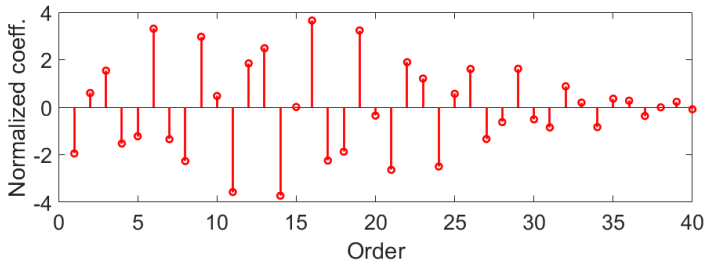


Speech
= e(n) * h(n)

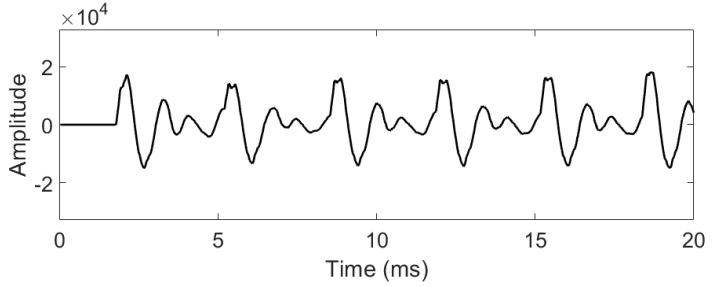




Source



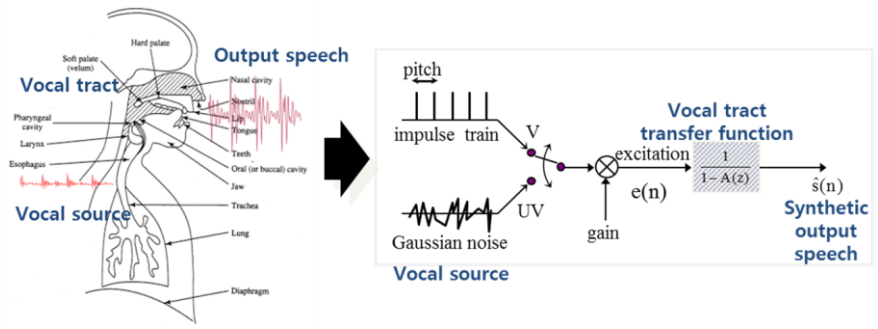
Filter

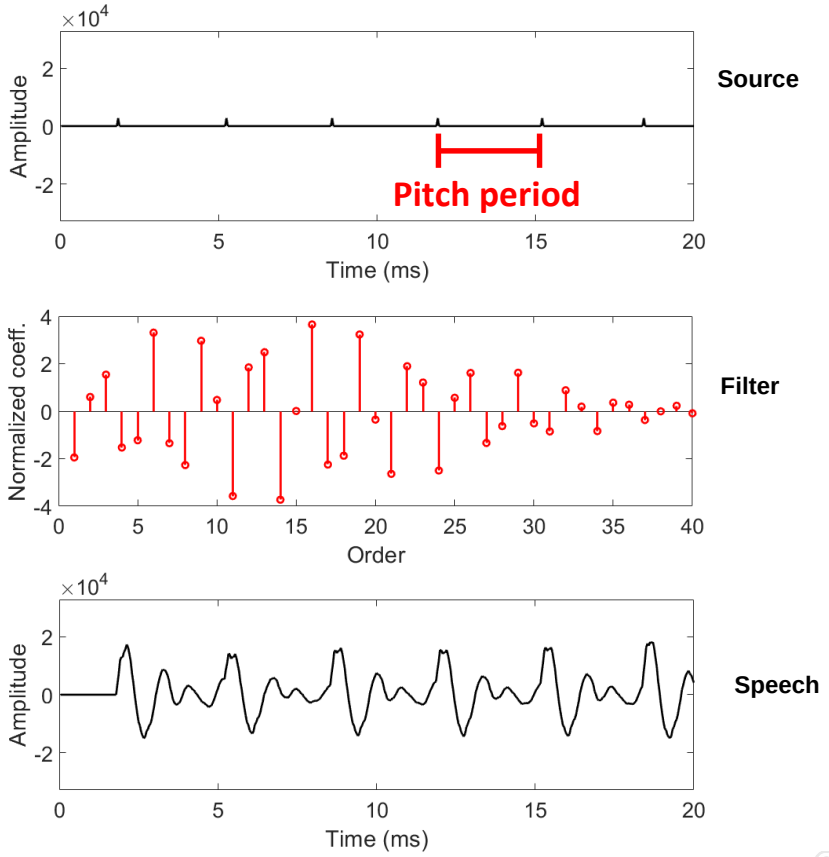


Speech

Parametric LPC synthesis

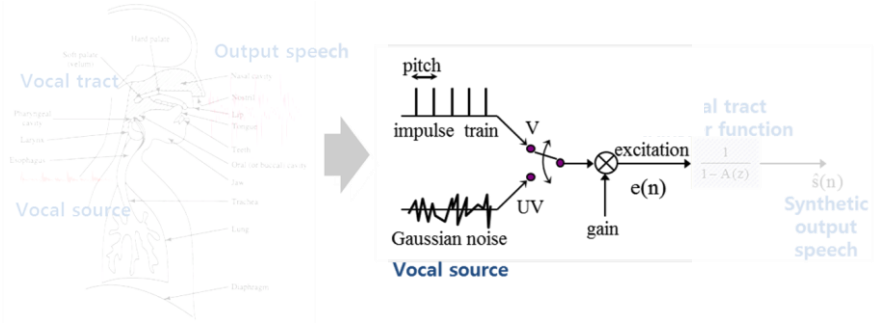
LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.





Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.

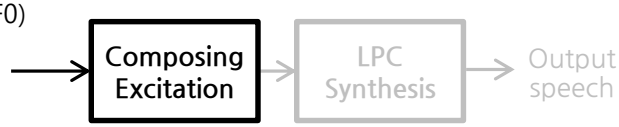


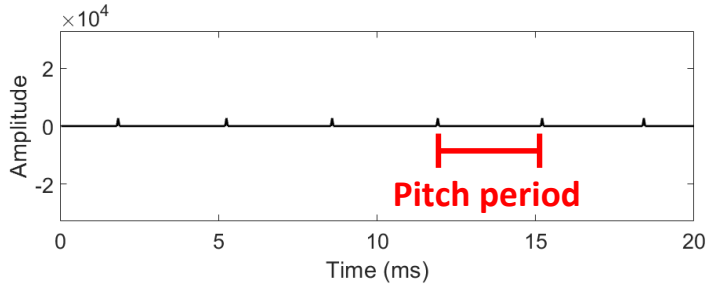
Excitation parameters

- Pitch period (or F0)
- Voicing flag
- Gain

Spectral parameters

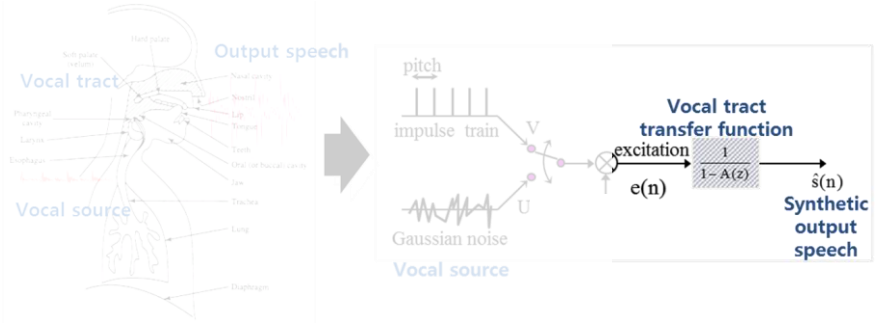
- LP coefficients





Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.

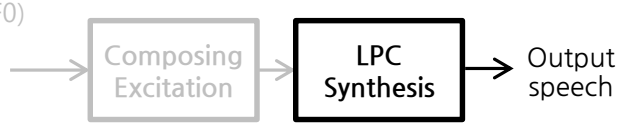


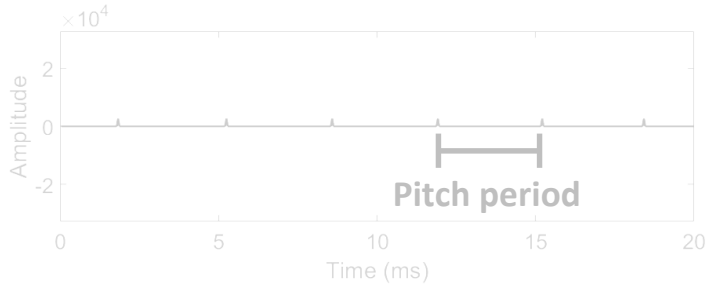
Excitation parameters

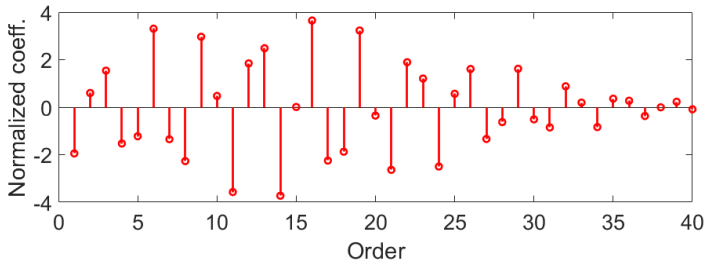
- Pitch period (or F0)
- Voicing flag
- Gain

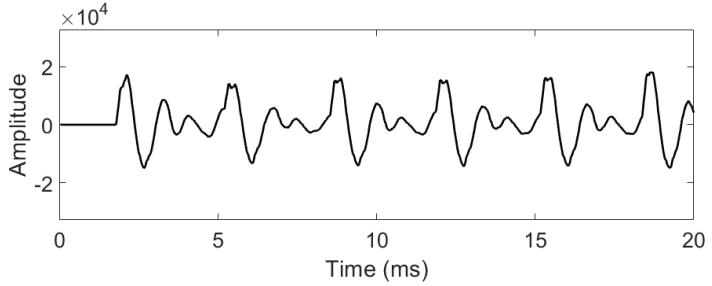
Spectral parameters

- LP coefficients



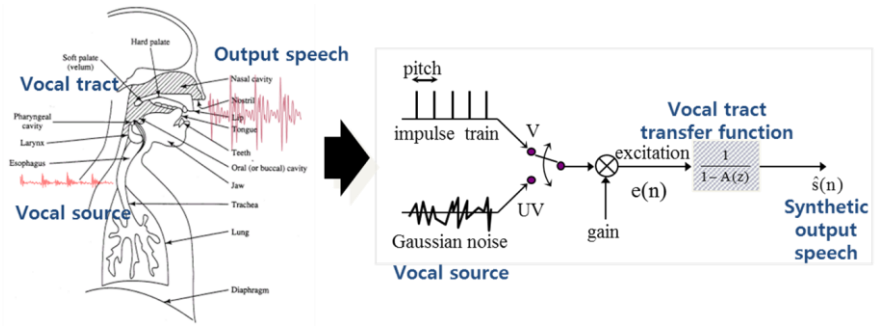






Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.



Excitation parameters

- Pitch period (or F0)
- Voicing flag
- Gain

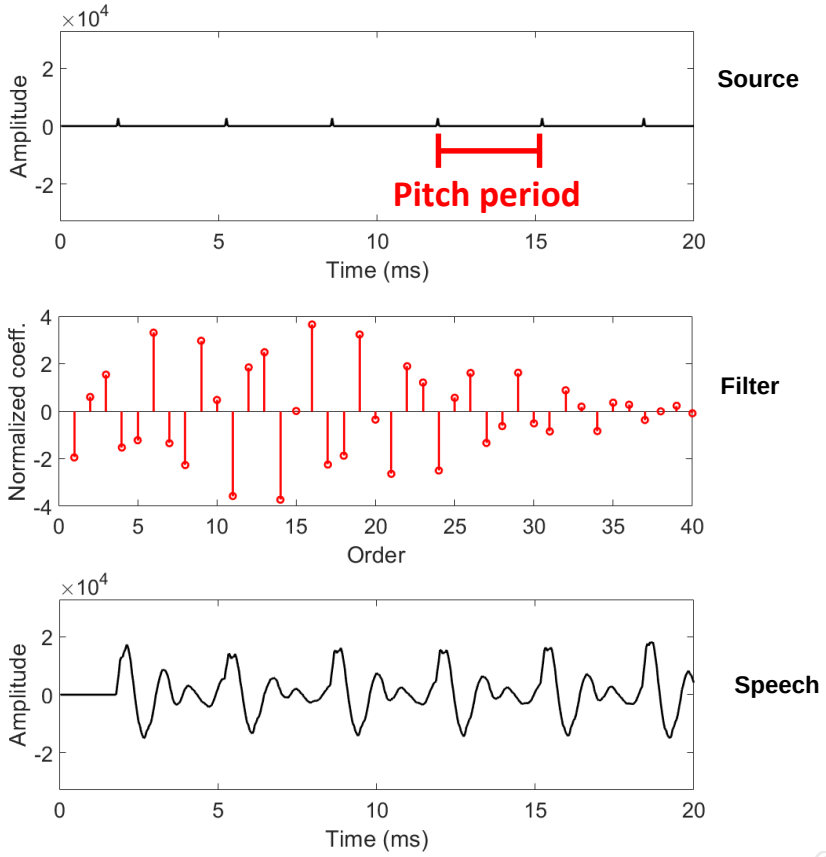
Spectral parameters

- LP coefficients

Composing Excitation → LPC Synthesis → Output speech

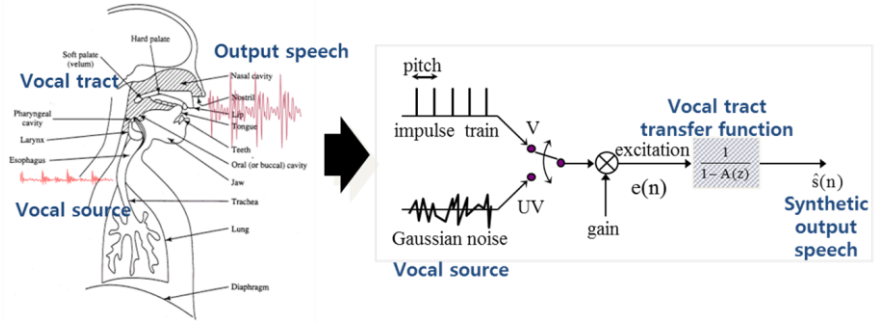
Acoustic parameters

Vocoder synthesis



Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.



Excitation parameters

- Pitch period (or F0)
- Voicing flag
- Gain

Spectral parameters

- LP coefficients

Composing Excitation → LPC Synthesis → Output speech

Acoustic parameters

Vocoder synthesis

Recorded speech

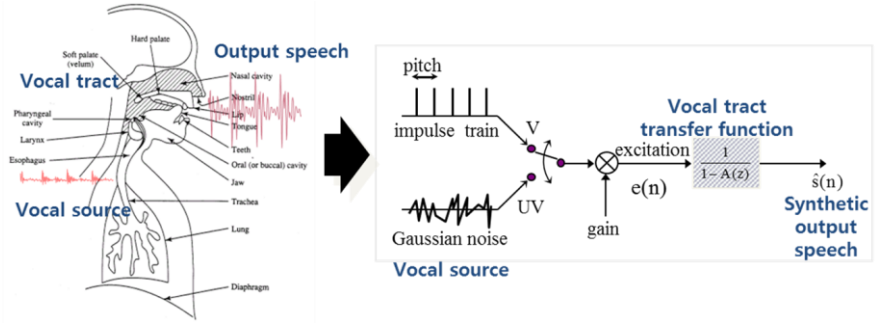
Generated speech

Low-tone

High-tone

Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.



Excitation parameters

- Pitch period (or F0)
- Voicing flag
- Gain

Spectral parameters

- LP coefficients

Composing Excitation → LPC Synthesis → Output speech

Acoustic parameters

Vocoder synthesis

Recorded speech

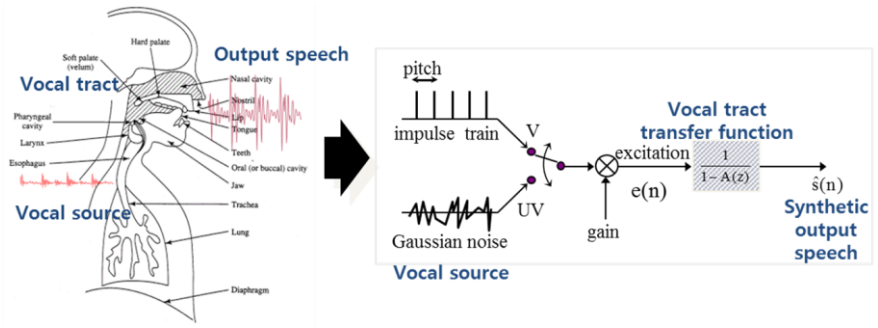
Generated speech

Low-tone

High-tone

Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.



Excitation parameters

- Pitch period (or F0)
- Voicing flag
- Gain

Spectral parameters

- LP coefficients

Composing Excitation → LPC Synthesis → Output speech

Acoustic parameters

Vocoder synthesis

Recorded speech

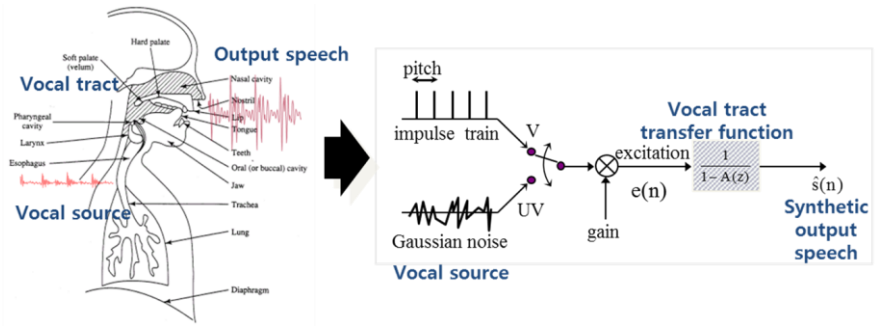
Generated speech

Low-tone

High-tone

Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.



Excitation parameters

- Pitch period (or F0)
- Voicing flag
- Gain

Spectral parameters

- LP coefficients

Composing Excitation → LPC Synthesis → Output speech

Acoustic parameters

Vocoder synthesis

Recorded speech

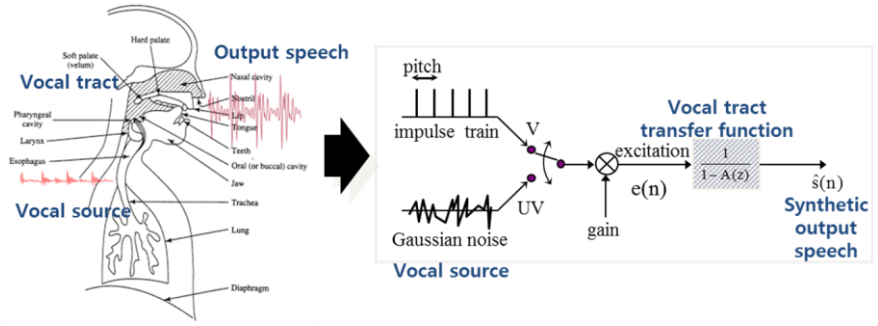
Generated speech

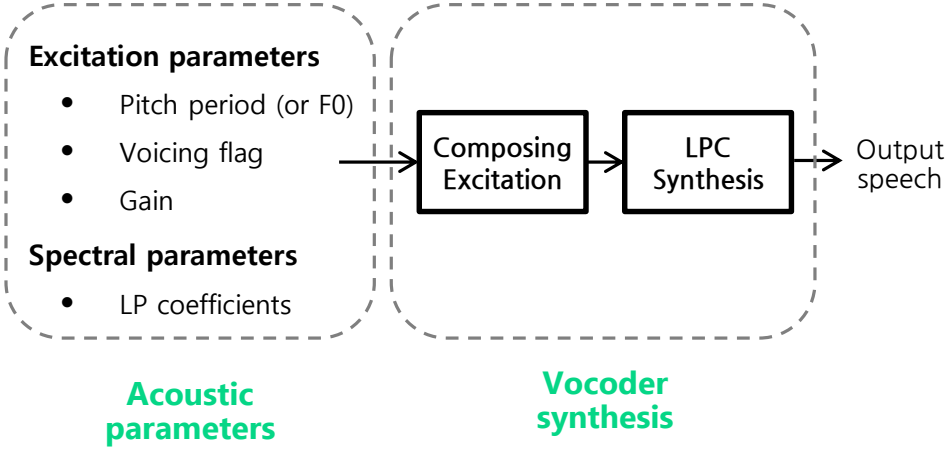
Low-tone

High-tone

Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.



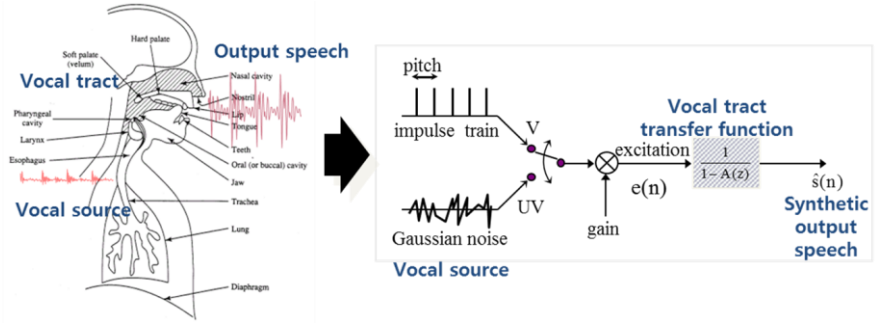


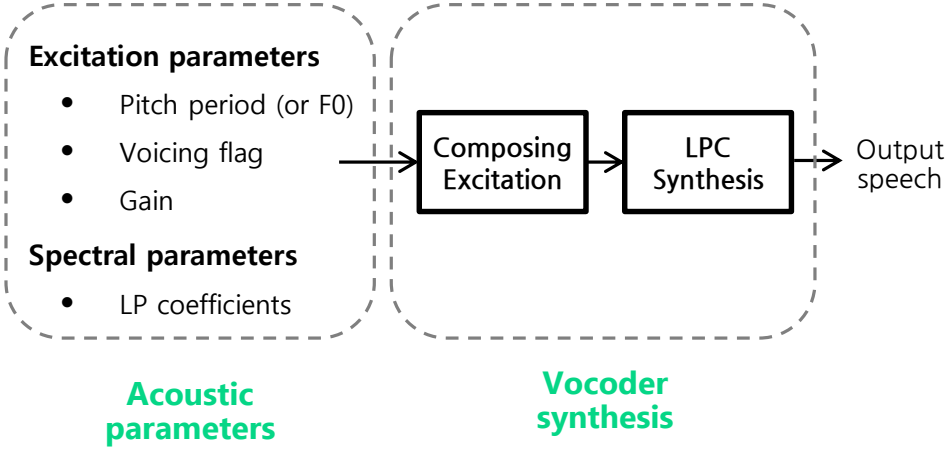
Spectral parameters

- How to extract LP coefficients ?
  - $\hat{s}(n) = \sum_{k=1}^{p} a(k)s(n-k)$
  - $e(n) = s(n) - \hat{s}(n) = s(n) - \sum_{k=1}^{p} a(k)s(n-k)$
- Minimizing mean square error
  - $\underset{a_k}{\operatorname{argmin}}\, E\left\{\left\|s(n) - \sum_{k=1}^{p} a(k)s(n-k)\right\|^2\right\}$
  - Levinson-Durbin recursion
- Parameterization
  - Line spectral frequency (LSF)
  - Mel-generalized cepstrum (MGC)
  - Mel-spectrum

Parametric LPC synthesis

LP coefficient 와 approximated excitation 을 이용해서 **음성**을 만들 수 있습니다.





Excitation parameters

- Approximation methods
  - Pulse or noise (PoN)
    - Pitch period, voicing flag, gain
  - Mixed excitation (STRAIGHT, WORLD)
    - Pitch period, voicing flag, gain
    - Band aperiodicity
  - Time-frequency trajectory excitation (TFTE)
    - Pitch period, voicing flag, gain
    - Slowly evolving waveform
    - Rapidly evolving waveform

Summary

음성 개념 1: Pitch period (or F0), formant

음성 개념 2: Speech production model, linear prediction

음성 개념 3: Parametric LPC vocoder

CLOVA

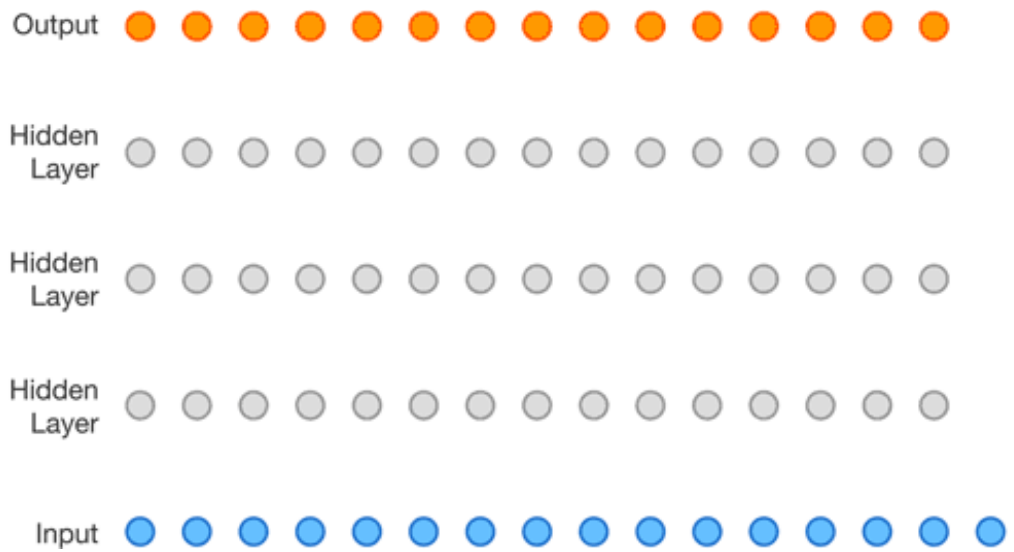
Q / A

Vocoding model

Autoregressive WaveNet vocoder

WaveNet synthesis

Neural network 로 **sample** 단위의 **음성** 신호를 추정할 수 있습니다.



현재 음성 신호를 예측할 때 과거 음성 신호를 함께 사용합니다.
이러한 방법을 **Autoregressive Model** 라고 정의합니다.

출처: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio

WaveNet synthesis

중요하니... 이론을 좀 ..

WaveNet

- A. Van den Oord, et. al., "WaveNet; a generative model for raw audio," CoRR abs/1609.03499, 2016.
- The first TTS algorithm that generates signal with a sample-by-sample manner

Properties

- Turn regression task into classification task (Speech is quantized to 8 bits (256 classes))
- Directly predicts the distribution of next sample, given condition and previous samples
- Maximize likelihood
  - $p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t | x_1, \cdots, x_{t-1})$

Key features

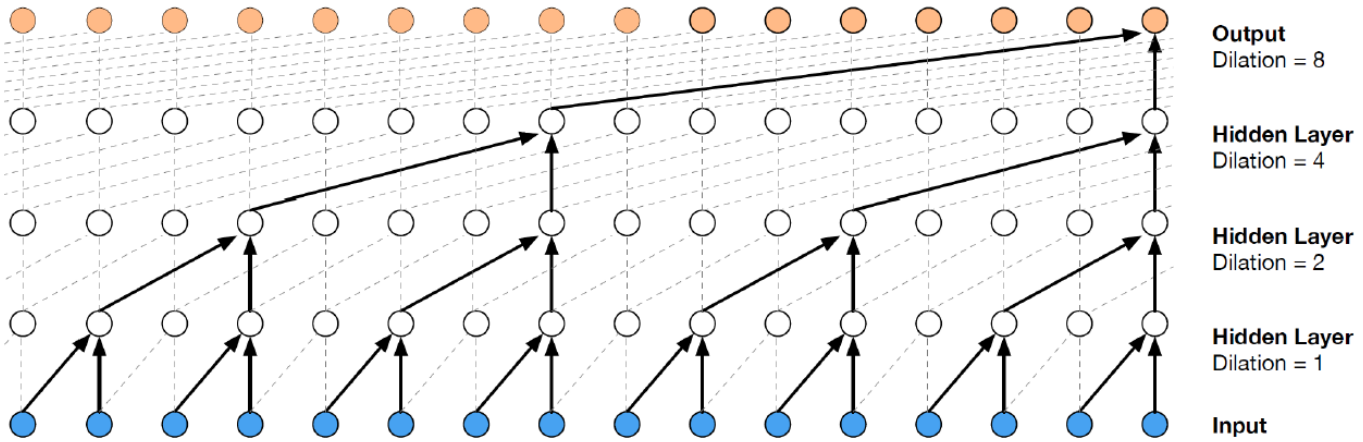
- Dilated causal convolutions
- Softmax distribution
- Gated activation units
- Residual and skip connections
- Conditional WaveNets

WaveNet synthesis

중요하니... 이론을 좀 ..

Dilated causal convolution

- Stacked dilated convolution: 1, 2, 4, 8, 16, ...



Softmax distributions

- 8 bit (256 level) mu-law companding transformation
  - $f(x_t) = sign(x_t)\frac{\ln(1+\mu|x_t|)}{\ln(1+\mu)}$

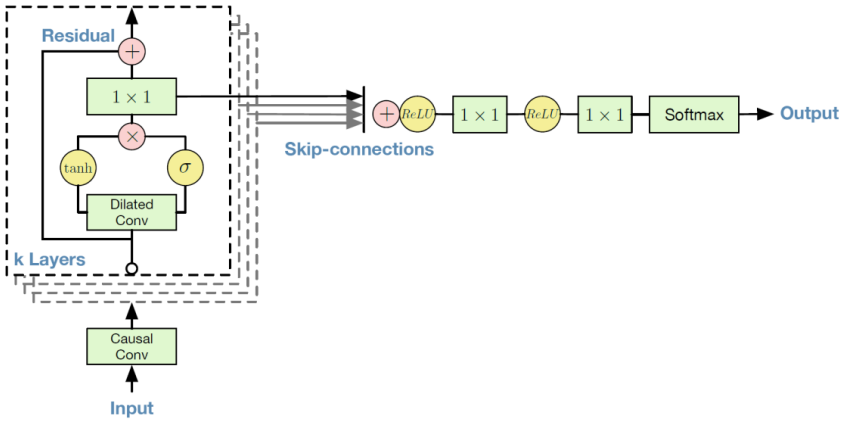
WaveNet synthesis

중요하니... 이론을 좀 ..

Gated activation units

- $\mathbf{z} = \tanh(W_{f,k} * \mathbf{x}) \odot \delta(W_{g,k} * \mathbf{x})$

Residual and skip connections



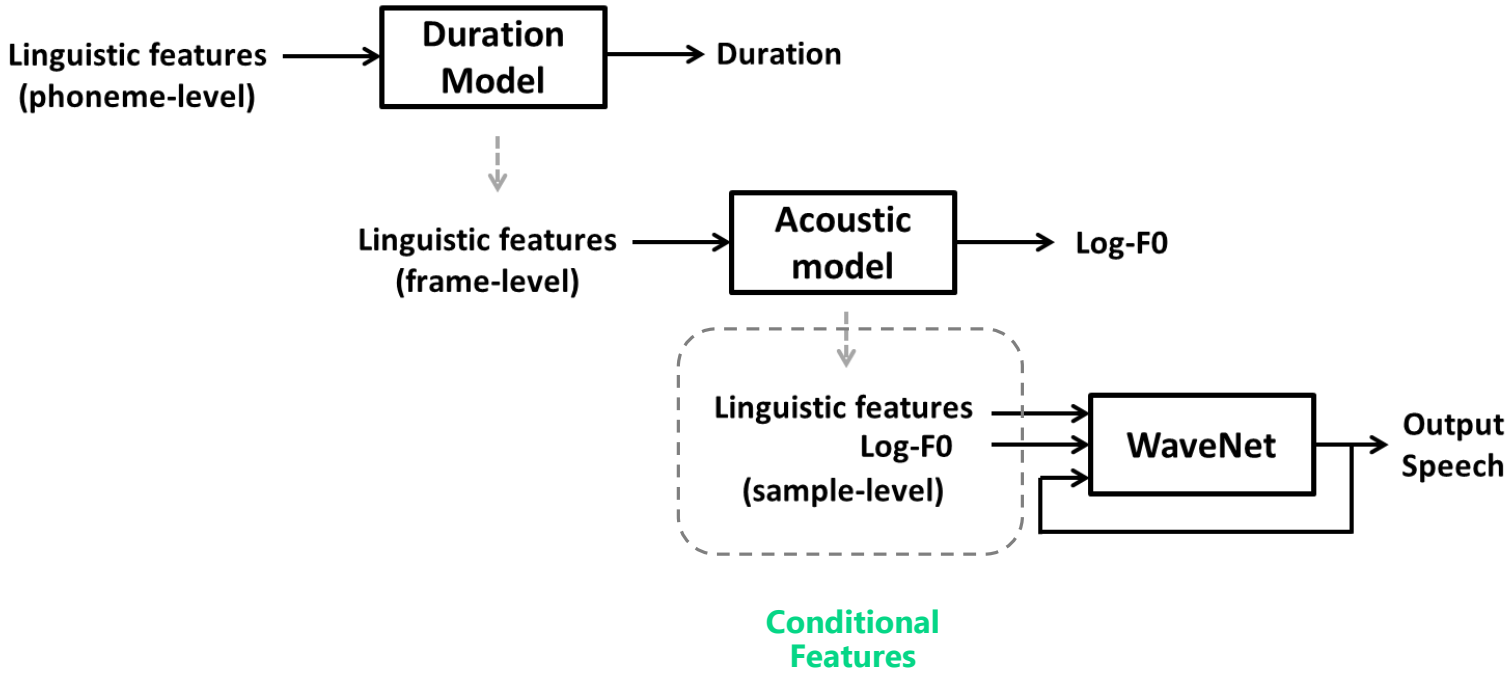
Conditional WaveNets

- $p(\mathbf{x}|\mathbf{h}) = \prod_{t=1}^{T} p(x_t|x_1, \cdots, x_{t-1}, \mathbf{h})$
- $\mathbf{z} = \tanh(W_{f,k} * \mathbf{x} + \boldsymbol{V}_{\boldsymbol{f},\boldsymbol{k}}^{\boldsymbol{T}}\mathbf{h}) \odot \delta(W_{g,k} * \mathbf{x} + \boldsymbol{V}_{\boldsymbol{g},\boldsymbol{k}}^{\boldsymbol{T}}\mathbf{h})$

WaveNet synthesis

End-to-end 는 아닙니다만 ..

처음에는 Vocoder 모델이 아니라 **End-to-end TTS 모델**로 사용되었습니다.



출처: A. Van Den Oord, et al., "WaveNet: A generative model for raw audio," *CoRR abs/1609.03499*, 2016.

WaveNet synthesis

Input Condition 으로 **Acoustic Parameter** 를 넣어줘야 비로소 **Vocoder** 가 됩니다.

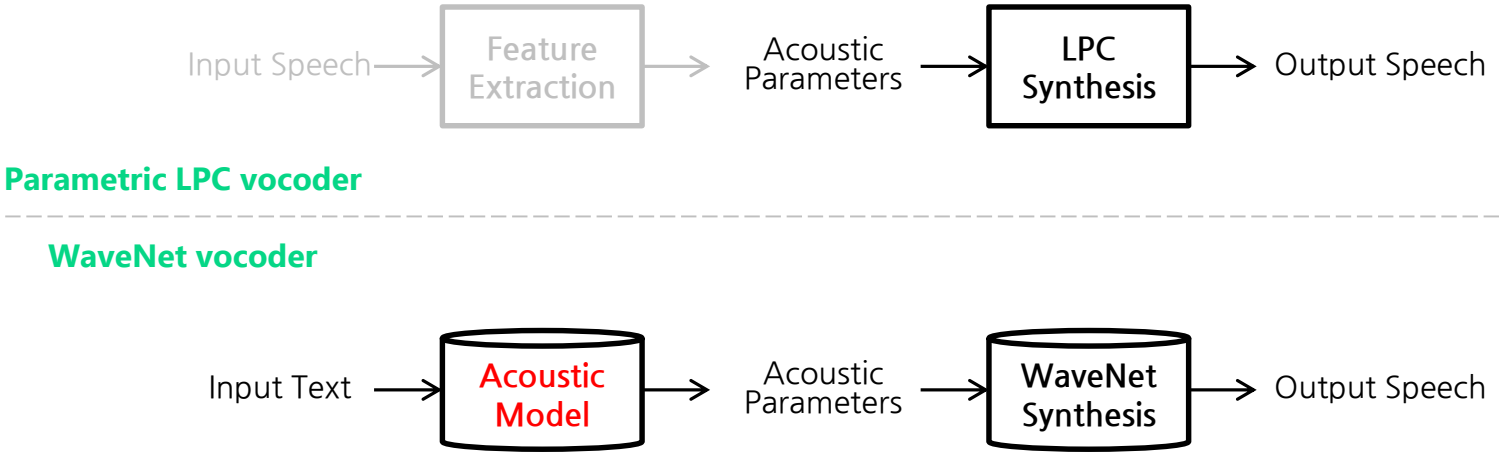
Parametric LPC vocoder

WaveNet vocoder

출처: A. Tamamory, et. al., “Speaker-dependent WaveNet vocoder,” in *Proc. Interspeech*, 2017.

WaveNet synthesis

Input Condition 으로 **Acoustic Parameter** 를 넣어줘야 비로소 **Vocoder** 가 됩니다.



Tacotron 2

출처: J. Shen, et. al., “Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions,” in *Proc. ICASSP*, 2018.

WaveNet synthesis

Parametric LPC Vocoder 보다 월등히 좋은 성능을 보여줍니다.

Table 1: *Comparative methods of waveform synthesis; spectrum envelop was extracted by STRAIGHT analysis.*

Comparative Method	Source of mel-cepstrum	Waveform Synthesis
Plain-MLSA	STFT	MLSA filter
STRAIGHT-MLSA	Spectrum envelop	MLSA filter
Plain-WaveNet	STFT	WaveNet
STRAIGHT-WaveNet	Spectrum envelop	WaveNet

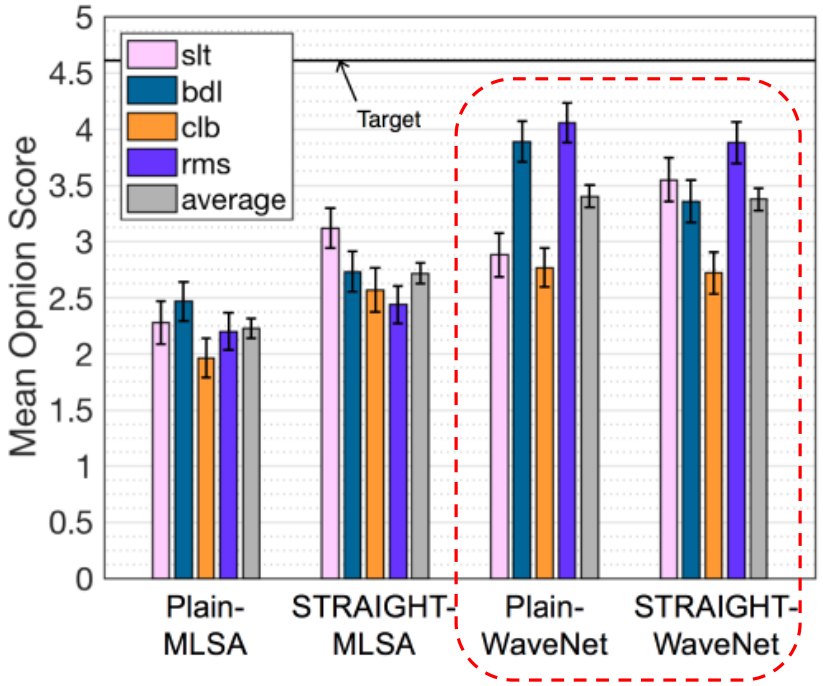


Figure 3: *Sound quality of synthesized speech*

Training data: 1 hour per each speaker

출처: A. Tamamory, et. al., “Speaker-dependent WaveNet vocoder,” in *Proc. Interspeech*, 2017.

WaveNet synthesis

WaveNet 모델의 성능을 더 높일 수 있는 방법

Table 1: *Comparative methods of waveform synthesis; spectrum envelop was extracted by STRAIGHT analysis.*

Comparative Method	Source of mel-cepstrum	Waveform Synthesis
Plain-MLSA	STFT	MLSA filter
STRAIGHT-MLSA	Spectrum envelop	MLSA filter
Plain-WaveNet	STFT	WaveNet
STRAIGHT-WaveNet	Spectrum envelop	WaveNet

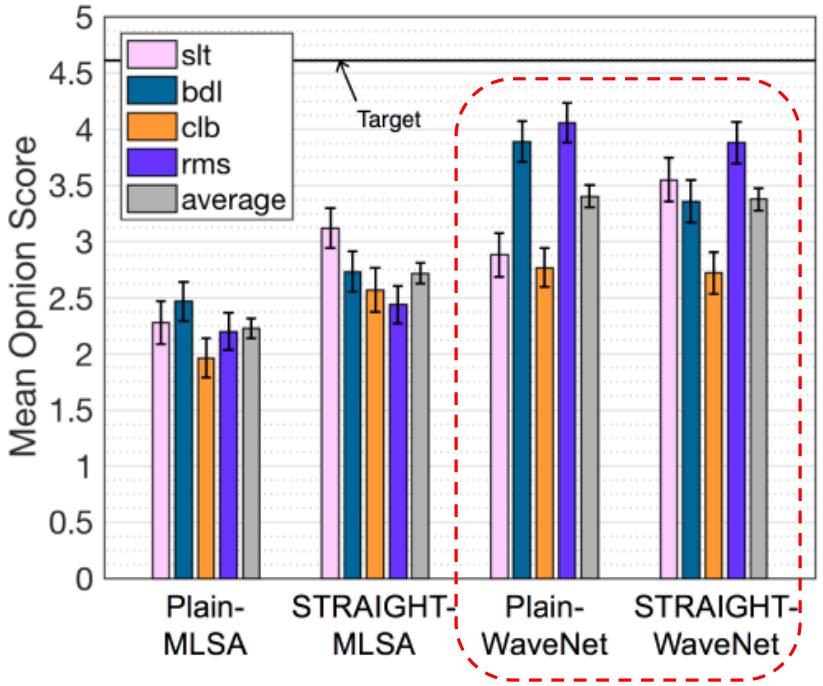


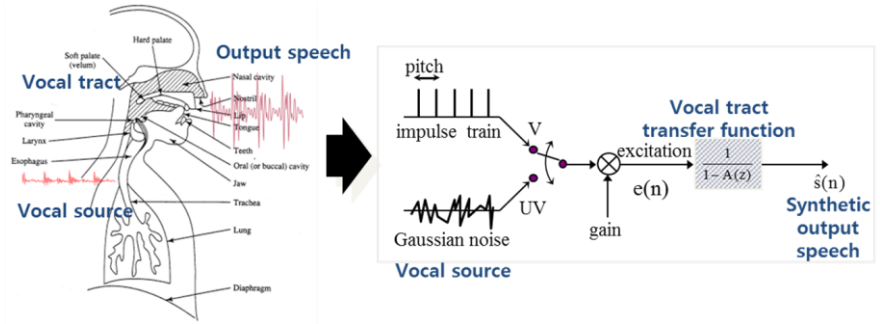
Figure 3: *Sound quality of synthesized speech*

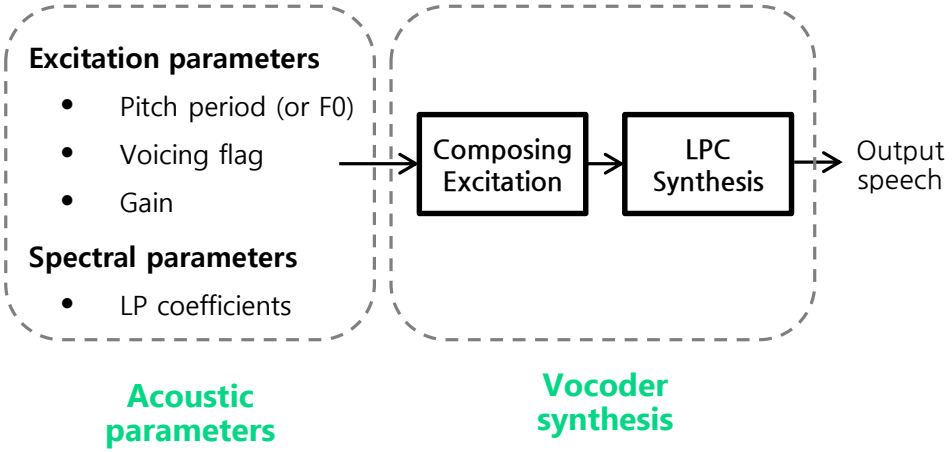
Training data: 1 hour per each speaker

출처: A. Tamamory, et. al., "Speaker-dependent WaveNet vocoder," in *Proc. Interspeech*, 2017.

Recall: Parametric LPC vocoder

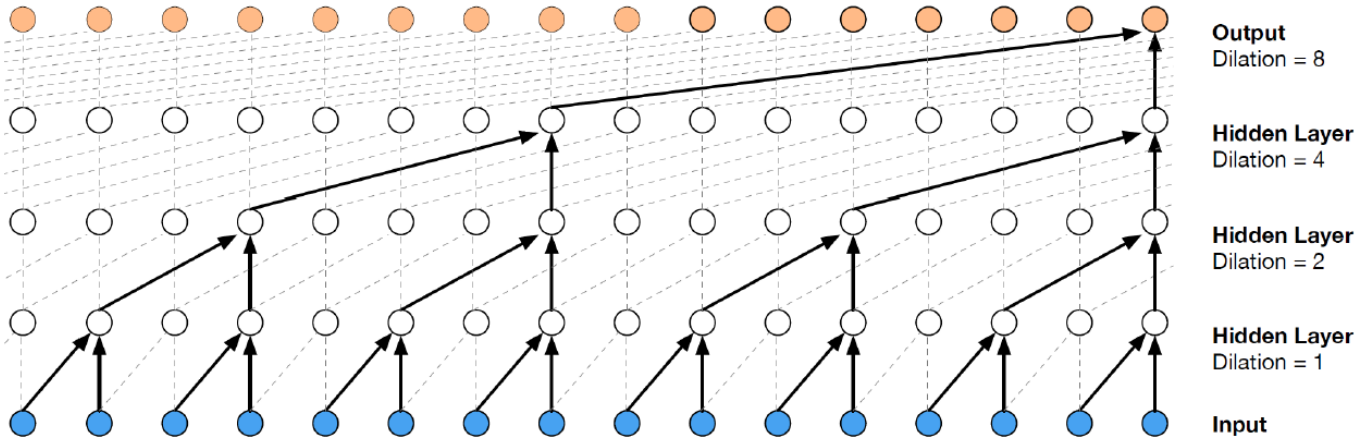
Excitation 신호를 추정하고 LPC Synthesis Filter를 이용해 음성을 만드는 방법





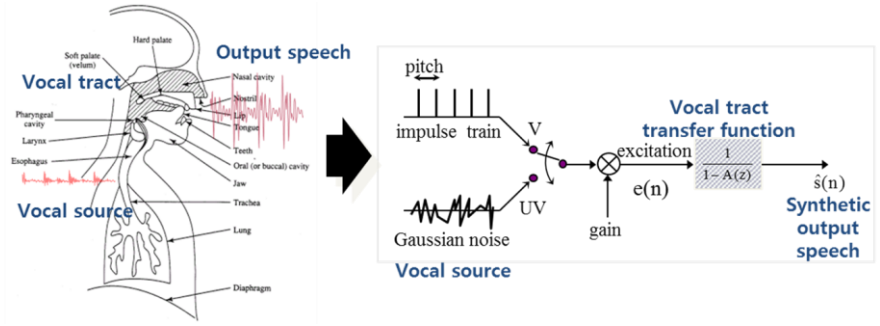
Recall: WaveNet vocoder

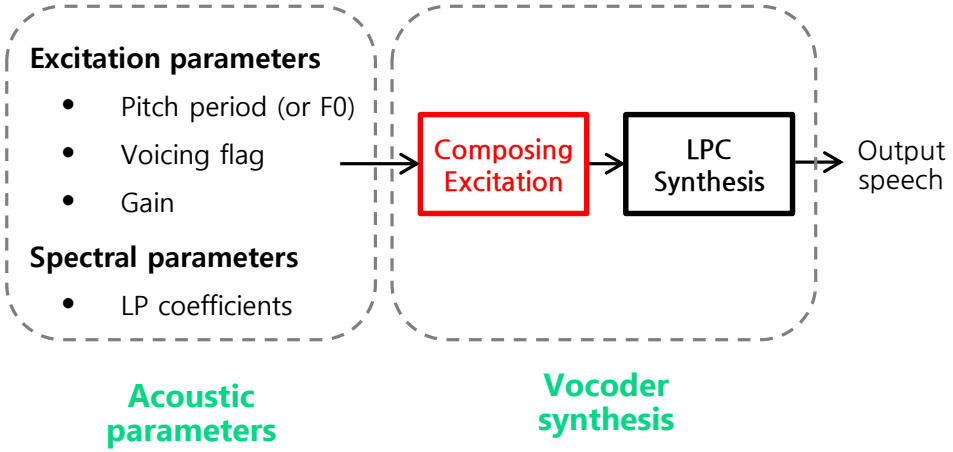
Time-domain 의 **음성** 샘플을 직접 추정하는 방법



Recall: WaveNet vocoder

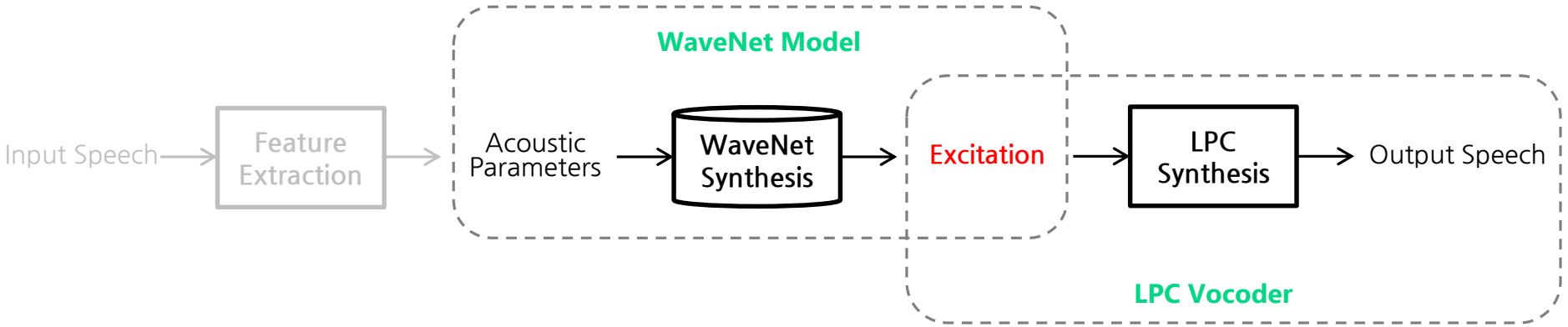
WaveNet 모델로 Time-domain 의 **Excitation** 샘플을 직접 추정한다면?





Neural excitation vocoder

합성음 품질을 더욱 높일 수 있다 !



Recorded speech

TTS + LPC vocoder

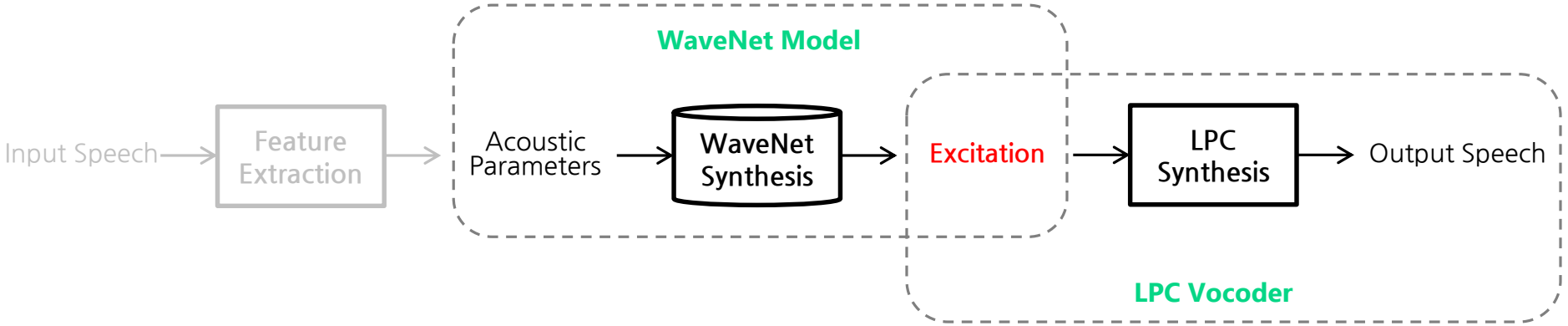
TTS + WaveNet vocoder

TTS + ExcitNet vocoder

출처: E. Song, et. al., “ExcitNet vocoder: A neural excitation model for parametric speech synthesis systems,” in *Proc. EUSIPCO*, 2019.

Neural excitation vocoder

합성음 품질을 더욱 높힐 수 있다 !

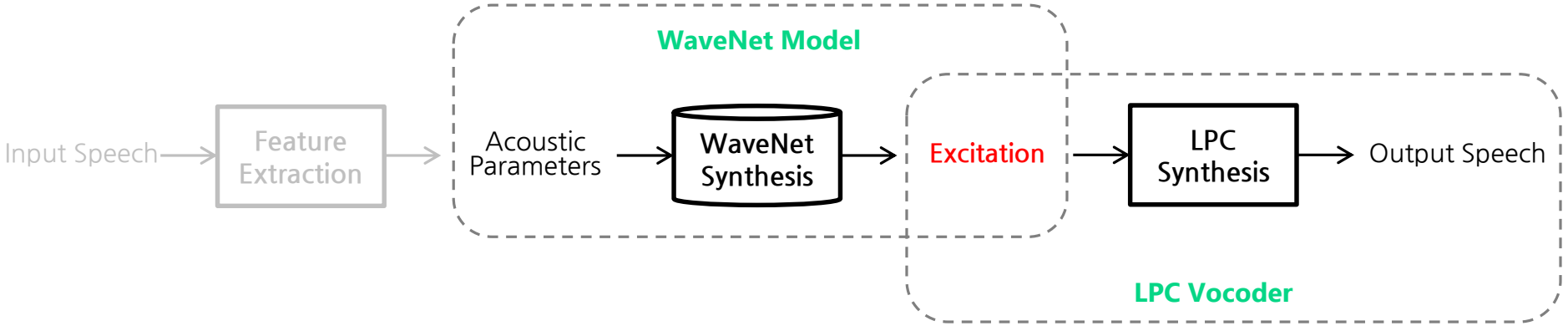


Recorded speech TTS + LPC vocoder TTS + WaveNet vocoder TTS + ExcitNet vocoder

출처: E. Song, et. al., "ExcitNet vocoder: A neural excitation model for parametric speech synthesis systems," in *Proc. EUSIPCO*, 2019.

Neural excitation vocoder

합성음 품질을 더욱 높힐 수 있다 !



Recorded speech

TTS + LPC vocoder

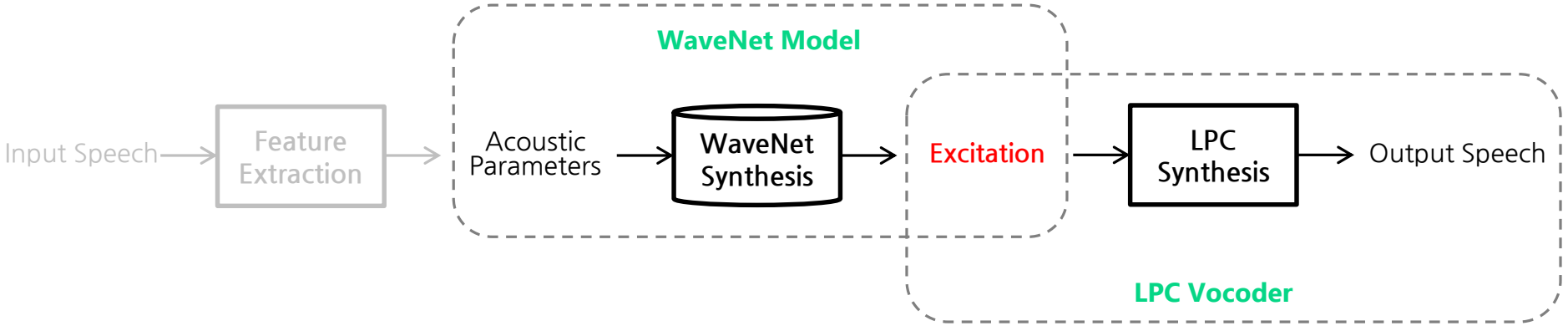
TTS + WaveNet vocoder

TTS + ExcitNet vocoder

출처: E. Song, et. al., “ExcitNet vocoder: A neural excitation model for parametric speech synthesis systems,” in *Proc. EUSIPCO*, 2019.

Neural excitation vocoder

합성음 품질을 더욱 높일 수 있다 !



Recorded speech

TTS + LPC vocoder

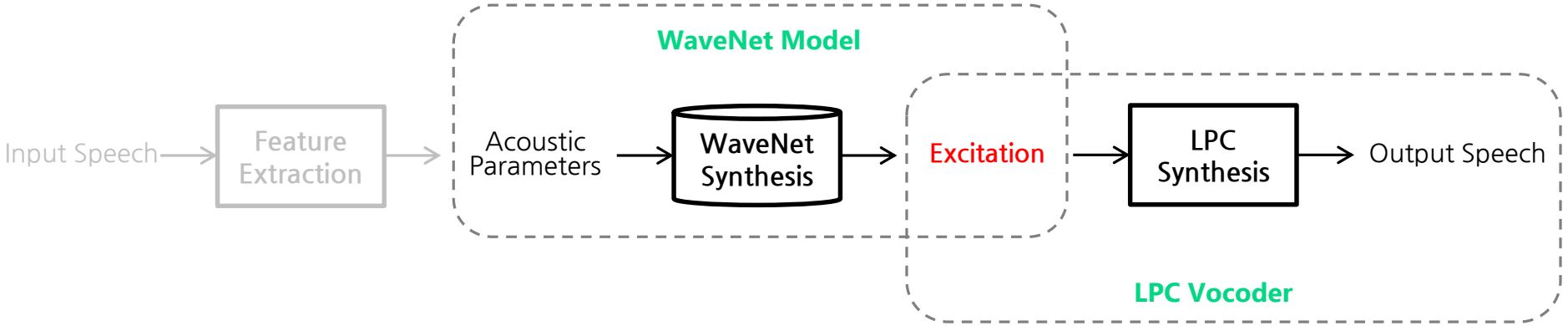
TTS + WaveNet vocoder

TTS + ExcitNet vocoder

출처: E. Song, et. al., “ExcitNet vocoder: A neural excitation model for parametric speech synthesis systems,” in *Proc. EUSIPCO*, 2019.

Neural excitation vocoder

합성음 품질을 더욱 높힐 수 있다 !



Recorded speech

TTS + LPC vocoder

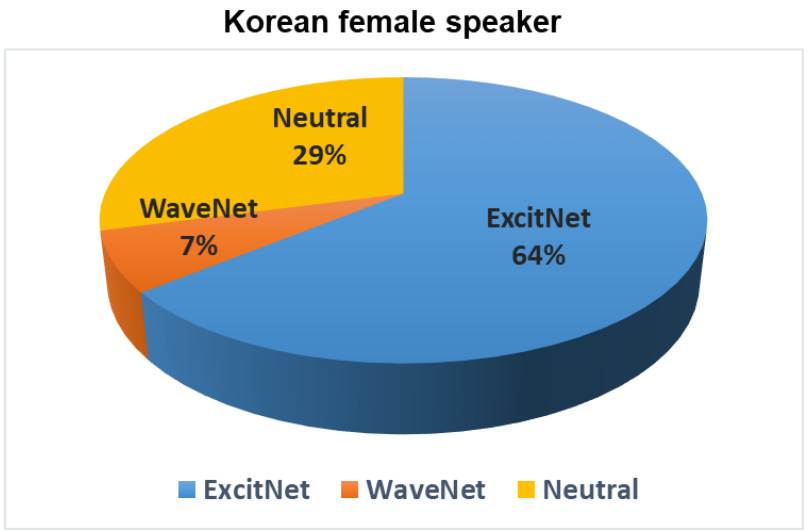
TTS + WaveNet vocoder

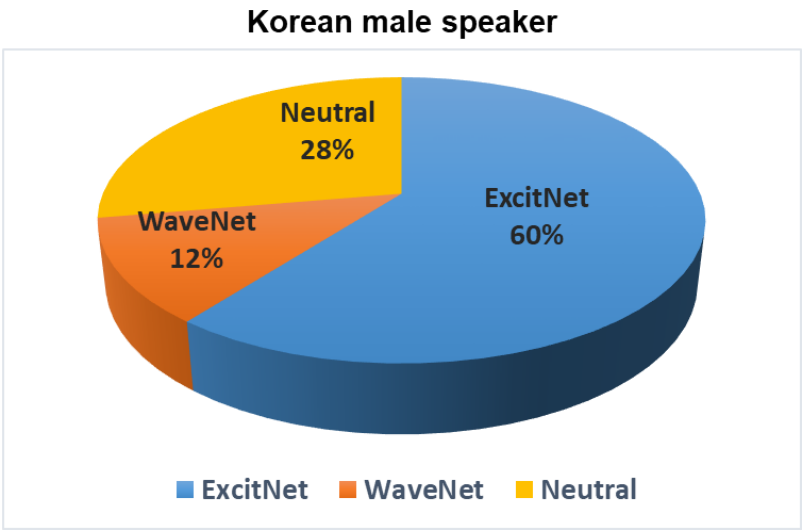
TTS + ExcitNet vocoder

출처: E. Song, et. al., “ExcitNet vocoder: A neural excitation model for parametric speech synthesis systems,” in *Proc. EUSIPCO*, 2019.

Neural excitation vocoder

합성음 품질을 더욱 높힐 수 있다 !





Recorded speech

TTS + LPC vocoder

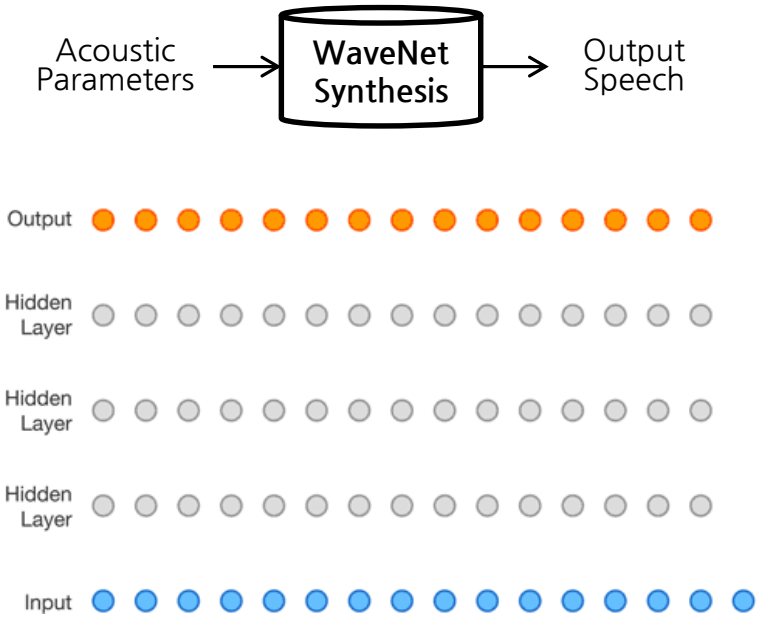
TTS + WaveNet vocoder

TTS + ExcitNet vocoder

출처: E. Song, et. al., “ExcitNet vocoder: A neural excitation model for parametric speech synthesis systems,” in *Proc. EUSIPCO*, 2019.

Summary

WaveNet Vocoder 를 꼭 기억해 주세요!



Autoregressive WaveNet vocoder

- Sample-by-sample generation
  - $p(\mathbf{x}|\mathbf{h}) = \prod_{t=1}^{T} p(x_t|x_1, \cdots, x_{t-1}, \mathbf{h})$
  - $\mathbf{h}$: Conditional acoustic parameter

Neural excitation vocoder

- WaveNet + LPC synthesis
  - GlottNet, ExcitNet, LP-WaveNet ...

Similar approaches

- WaveRNN, SampleRNN vocoder
  - RNN-based generation (cf. WaveNet: CNN)
  - LPCNet: WaveRNN + LPC synthesis

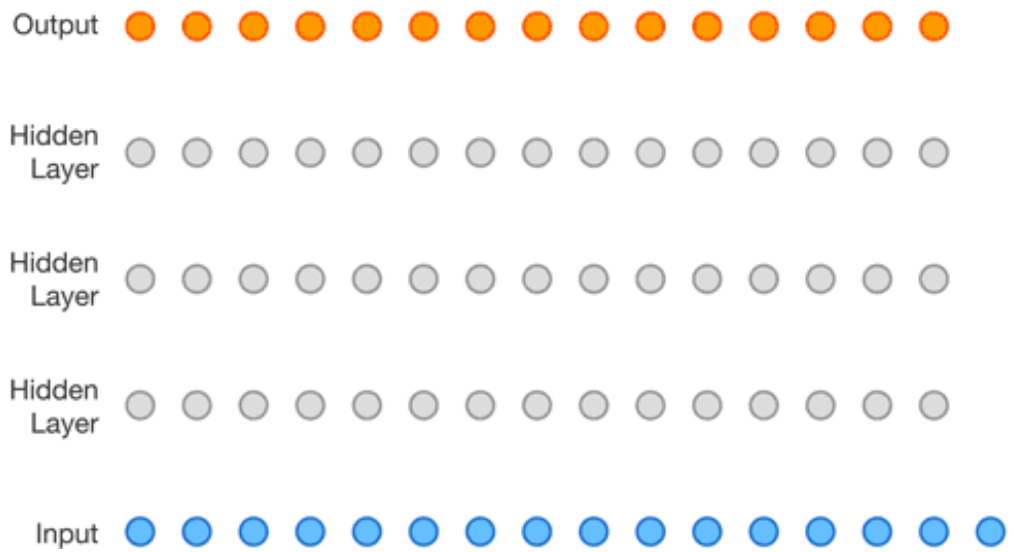
CLOVA

Vocoding model

Non-autoregressive WaveNet synthesis

Recall

현재 음성 신호를 예측할 때 **과거 음성** 신호를 함께 사용하는 방법: **Autoregressive** Model

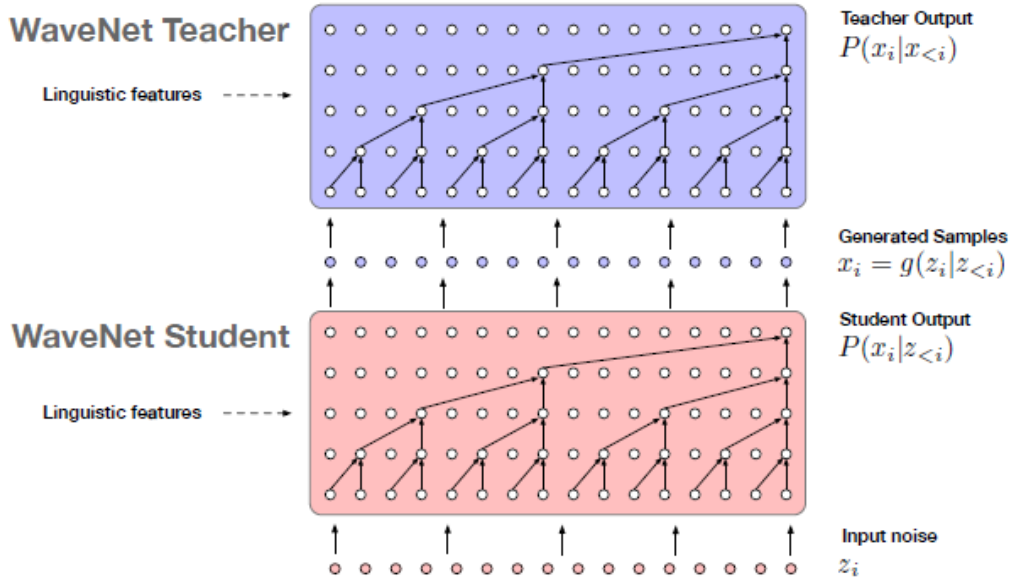


Autoregressive Model 은 고품질의 음성을 생성할 수 있으나,
1초 음성을 만들 때 약 5분 정도의 시간이 소요된다는 치명적인 문제가 있습니다.

출처: https://deepmind.com/blog/article/wavenet-generative-model-raw-audio

Parallel WaveNet

음성 신호를 **Parallel** 방식으로 예측하는 방법: **Non-autoregressive** Model

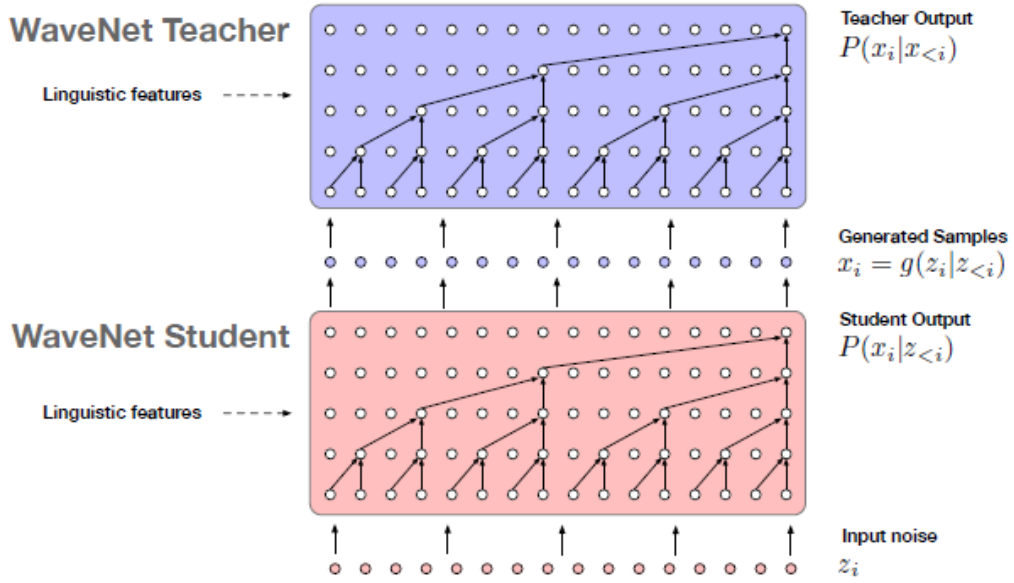


WaveNet 의 속도 문제를 해결하기 위해 제안된 방법이
Non-autoregressive 구조의 **Parallel WaveNet** 입니다.

출처: A. van den Oord, et al., “Parallel WaveNet: Fast high-fidelity speech synthesis,” in *Proc. ICML*, 2018.

Parallel WaveNet

음성 신호를 **Parallel** 방식으로 예측하는 방법: **Non-autoregressive** Model

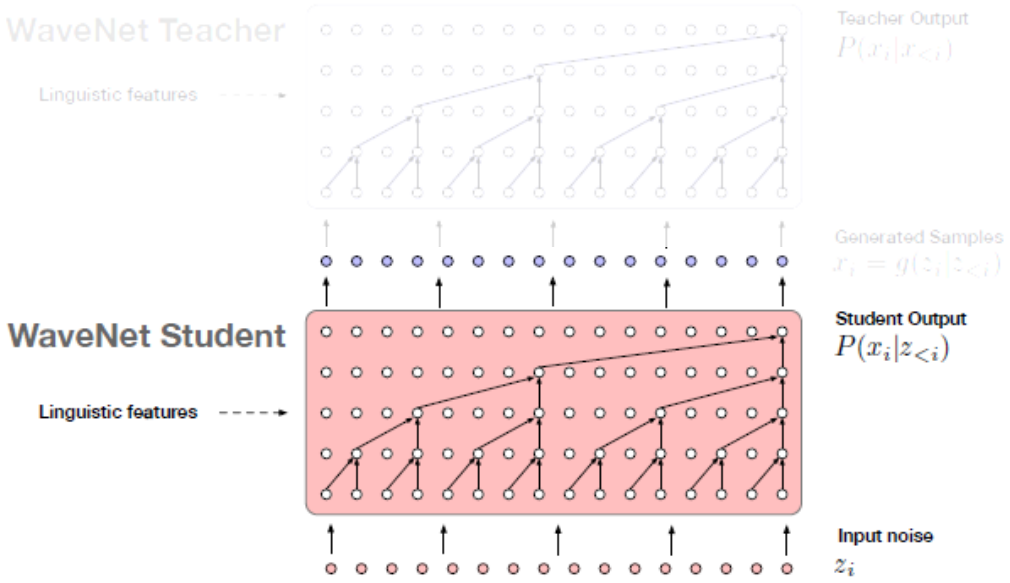


Autoregressive WaveNet (=Teacher) 모델의 확률 분포를
Non-autoregressive Parallel WaveNet (=Student) 모델이 배우도록 훈련합니다.

출처: A. van den Oord, et al., "Parallel WaveNet: Fast high-fidelity speech synthesis," in *Proc. ICML*, 2018.

Parallel WaveNet

음성 신호를 **Parallel** 방식으로 예측하는 방법: **Non-autoregressive** Model

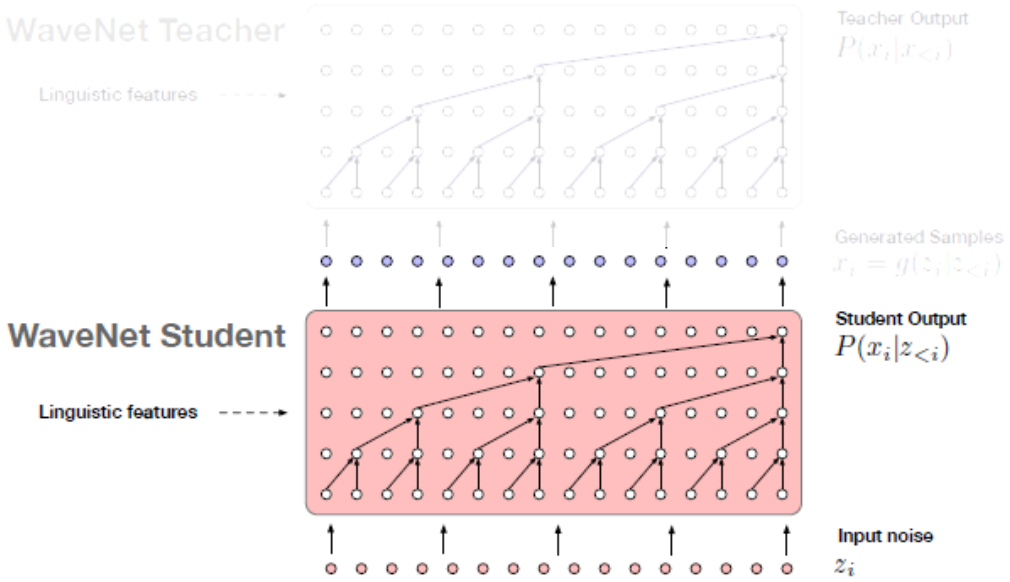


Non-autoregressive Parallel WaveNet 모델은
과거 음성을 사용하지 않으므로, 생성 속도에 제한이 없습니다.
(1초 음성을 약 0.02초 만에 생성 가능)

출처: A. van den Oord, et al., “Parallel WaveNet: Fast high-fidelity speech synthesis,” in *Proc. ICML*, 2018.

Parallel WaveNet

음성 신호를 **Parallel** 방식으로 예측하는 방법: **Non-autoregressive** Model

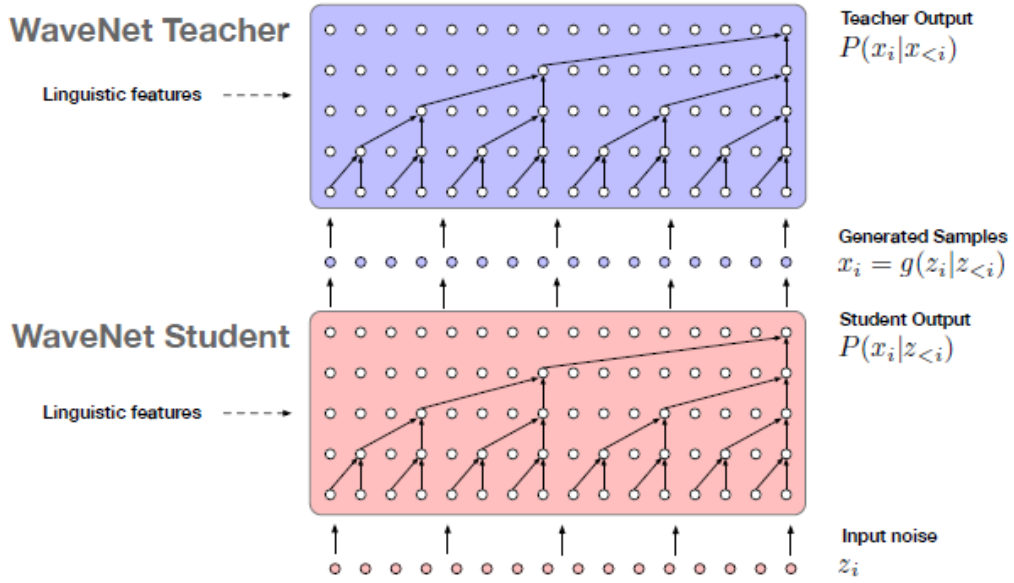


하지만 그만큼 합성음의 품질이 저하된다는 문제가 남아있습니다.

출처: A. van den Oord, et al., “Parallel WaveNet: Fast high-fidelity speech synthesis,” in *Proc. ICML*, 2018.

Parallel WaveNet

Autoregressive vs Non-autoregressive



Autoregressive

합성음 품질이 좋지만
생성 속도가 느리다

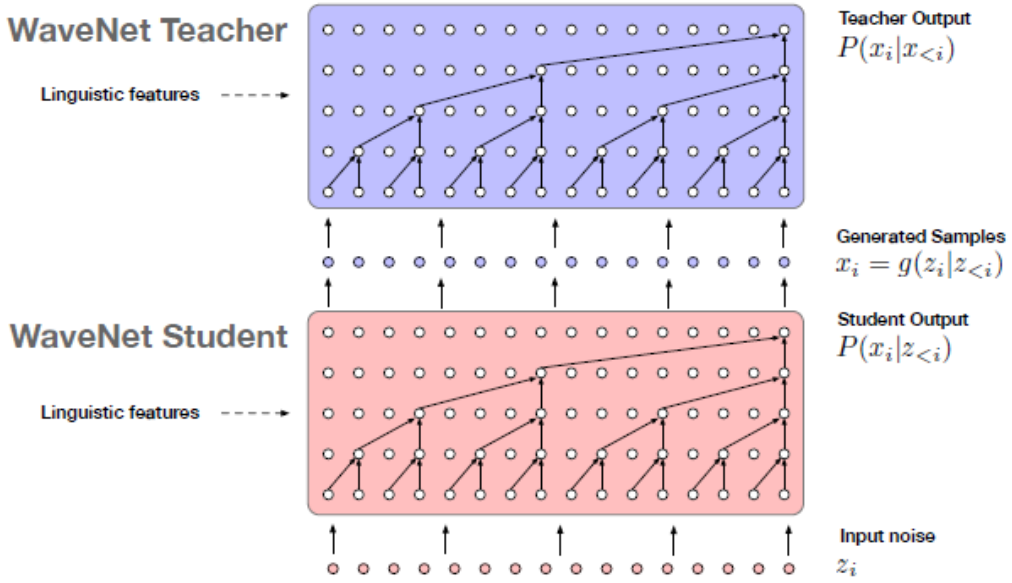
VS

Non-autoregressive

생성 속도가 빠르지만
학습이 어렵고
합성음 품질이 나쁘다

Parallel WaveNet

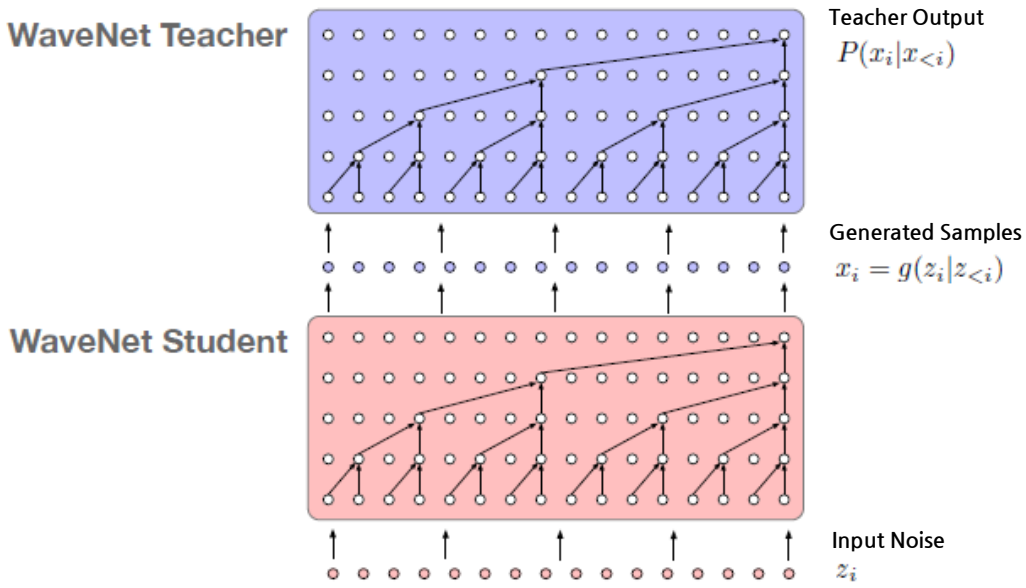
합성음 품질도 좋고, 생성속도도 빠른 WaveNet 은 없을까



Autoregressive		Non-autoregressive
합성음 품질이 좋지만 생성 속도가 느리다	VS	생성 속도가 빠르지만 학습이 어렵고 합성음 품질이 나쁘다

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,

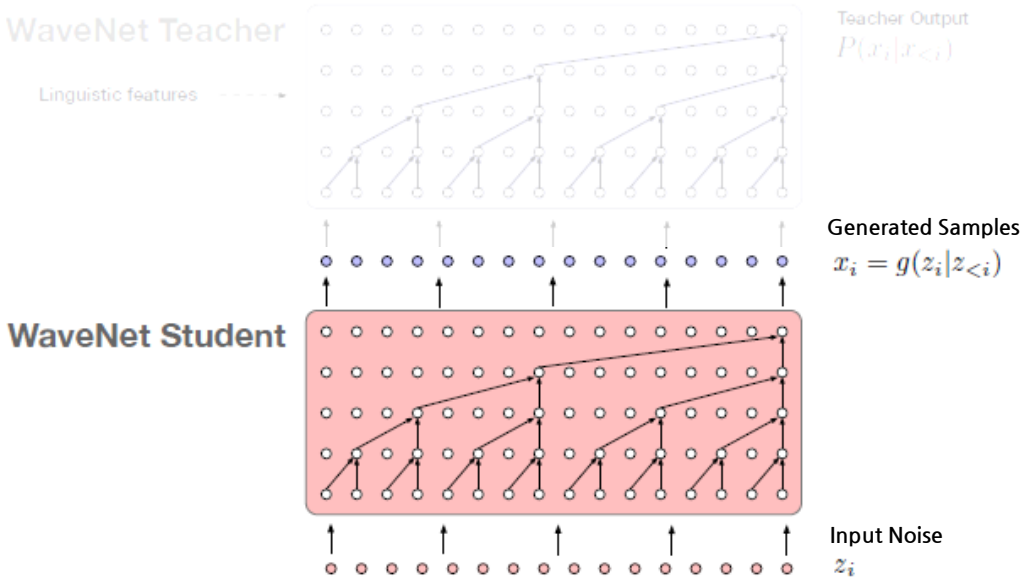


출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,

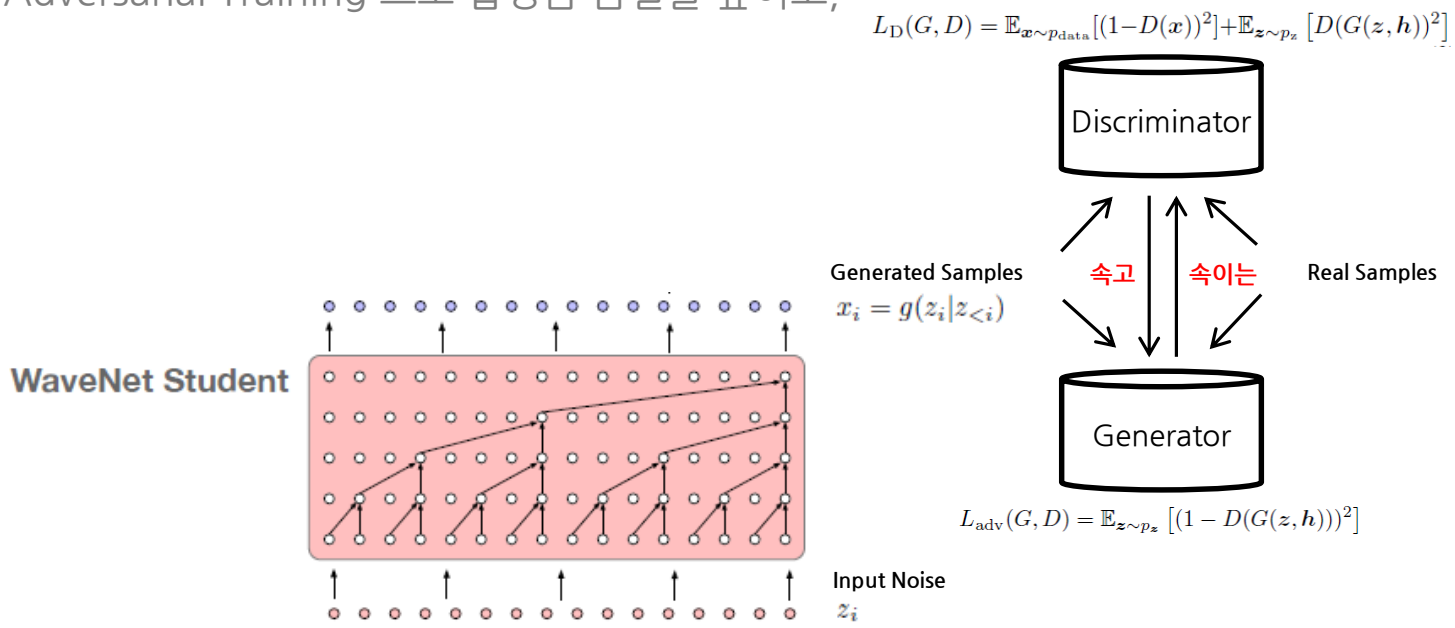
학습이 너무 어려우니까



출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

Parallel WaveGAN

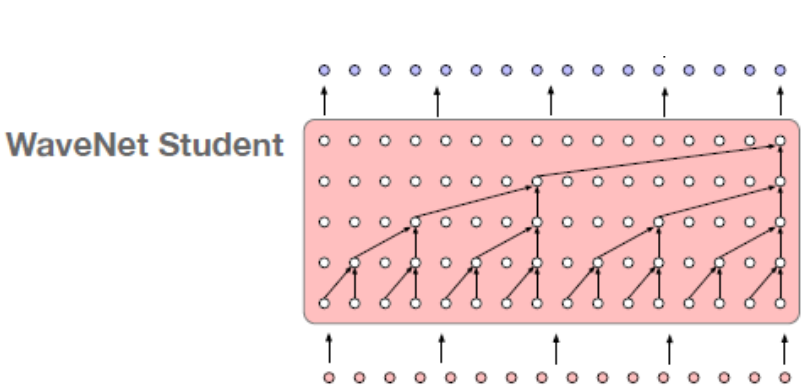
1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,



출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,



$$L_{\text{mr\_stft}}(G) = \frac{1}{M}\sum_{m=1}^{M} L_{\text{stft}}^{(m)}(G)$$

$$L_{\text{stft}}(G) = \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}},\boldsymbol{x}\sim p_{data}}\left[L_{\text{sc}}(\boldsymbol{x},\hat{\boldsymbol{x}}) + L_{\text{mag}}(\boldsymbol{x},\hat{\boldsymbol{x}})\right]$$

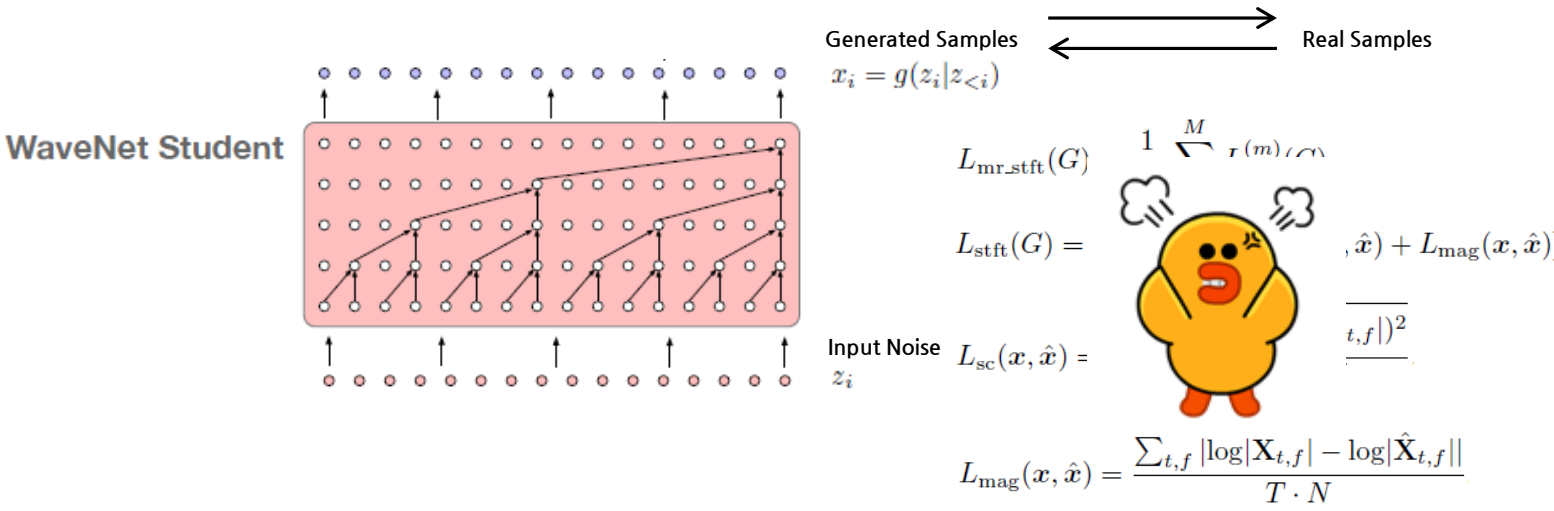
$$L_{\text{sc}}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sqrt{\sum_{t,f}(|\mathbf{X}_{t,f}| - |\hat{\mathbf{X}}_{t,f}|)^2}}{\sqrt{\sum_{t,f}|\mathbf{X}_{t,f}|^2}}$$

$$L_{\text{mag}}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sum_{t,f}|\log|\mathbf{X}_{t,f}| - \log|\hat{\mathbf{X}}_{t,f}||}{T\cdot N}$$

출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,



출처: R. Yamamoto, et al., "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. ICASSP*, 2020.

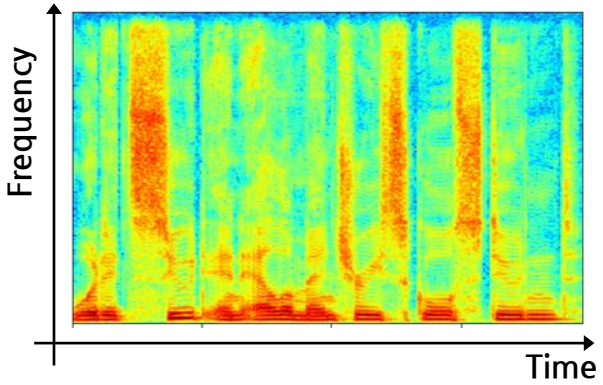
Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,

STFT (short-time Fourier transform)?

짧은 구간의 음성 신호를 주파수 축에서 표현 한 것.

시간 축으로 붙여서 2D 신호로 만든 것이 다음과 같은 Spectrogrm

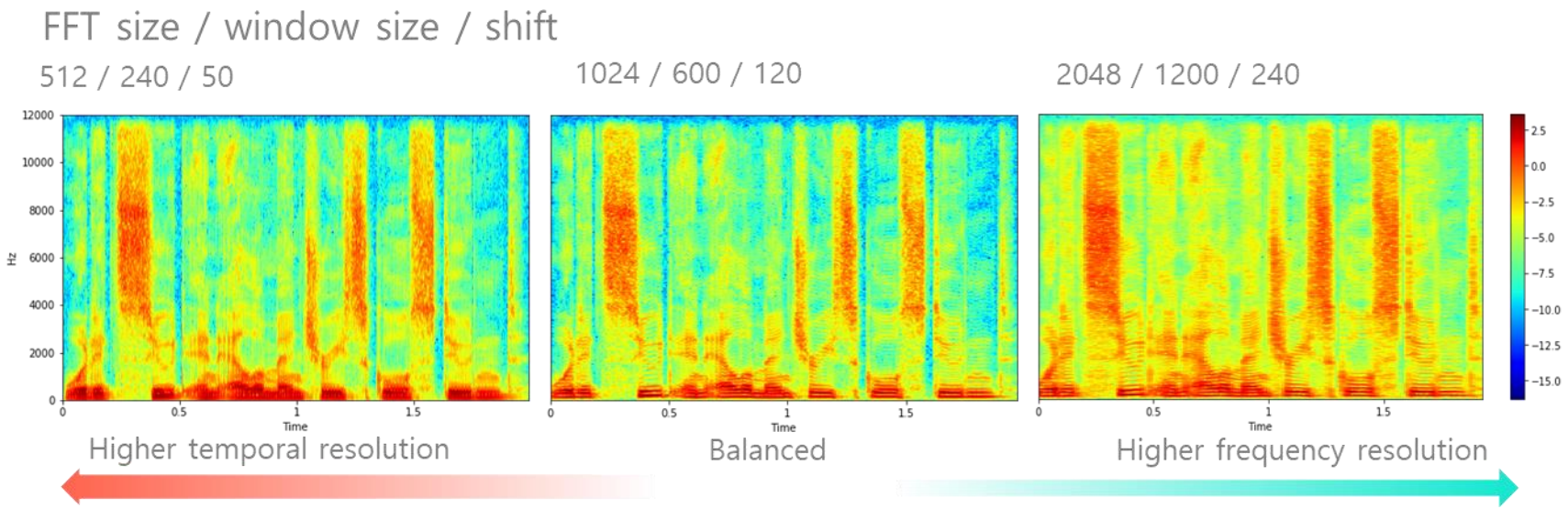


출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,

   시간-주파수 축에서 해상도가 다른 여러 개의 Loss 들의 평균이다.



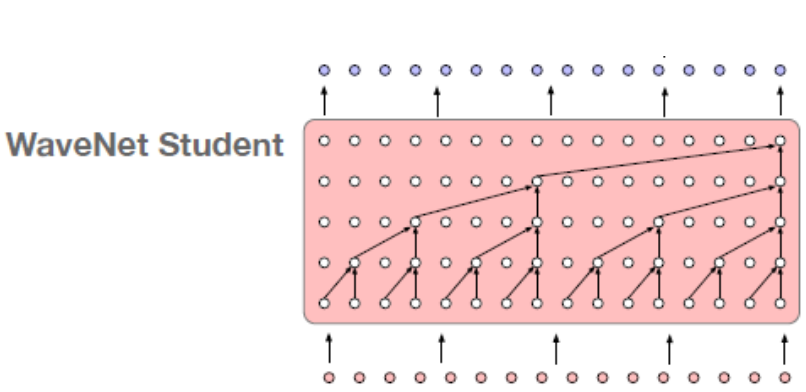
출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,

시간-주파수 축에서 해상도가 다른 여러 개의 Loss 들의 평균이다.

이때, Loss 는 두가지로 구성되는데



$$L_{\text{mr\_stft}}(G) = \frac{1}{M}\sum_{m=1}^{M} L_{\text{stft}}^{(m)}(G)$$

$$L_{\text{stft}}(G) = \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}},\boldsymbol{x}\sim p_{data}}\left[L_{\text{sc}}(\boldsymbol{x},\hat{\boldsymbol{x}}) + L_{\text{mag}}(\boldsymbol{x},\hat{\boldsymbol{x}})\right]$$

$$L_{\text{sc}}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sqrt{\sum_{t,f}(|\mathbf{X}_{t,f}| - |\hat{\mathbf{X}}_{t,f}|)^2}}{\sqrt{\sum_{t,f}|\mathbf{X}_{t,f}|^2}}$$

$$L_{\text{mag}}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sum_{t,f}|\log|\mathbf{X}_{t,f}| - \log|\hat{\mathbf{X}}_{t,f}||}{T\cdot N}$$

출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

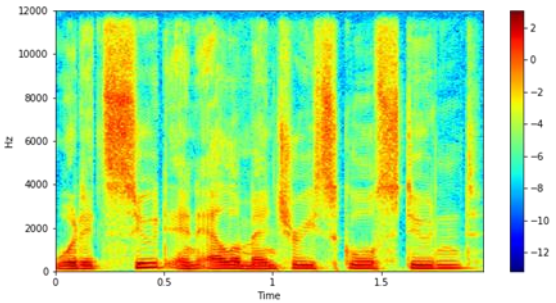
Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,

시간-주파수 축에서 해상도가 다른 여러 개의 Loss 들의 평균이다.

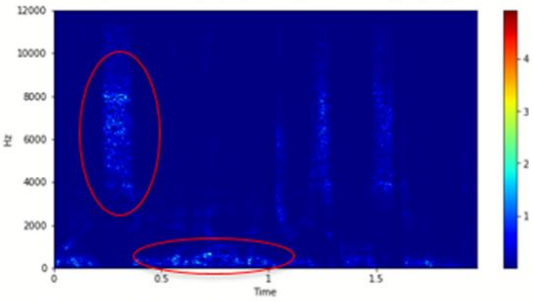
이때, Loss 는 두가지로 구성되는데

하나는 **에너지가 큰 구간**을 잡아내고



Real Spectrogram

$||\text{STFT}(x)| - |\text{STFT}(\hat{x})||$



SC penalizes large amplitude components

$$L_{\text{sc}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{\sqrt{\sum_{t,f}(|\mathbf{X}_{t,f}| - |\hat{\mathbf{X}}_{t,f}|)^2}}{\sqrt{\sum_{t,f}|\mathbf{X}_{t,f}|^2}}$$

출처: R. Yamamoto, et al., “Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram,” in *Proc. ICASSP*, 2020.

Parallel WaveGAN

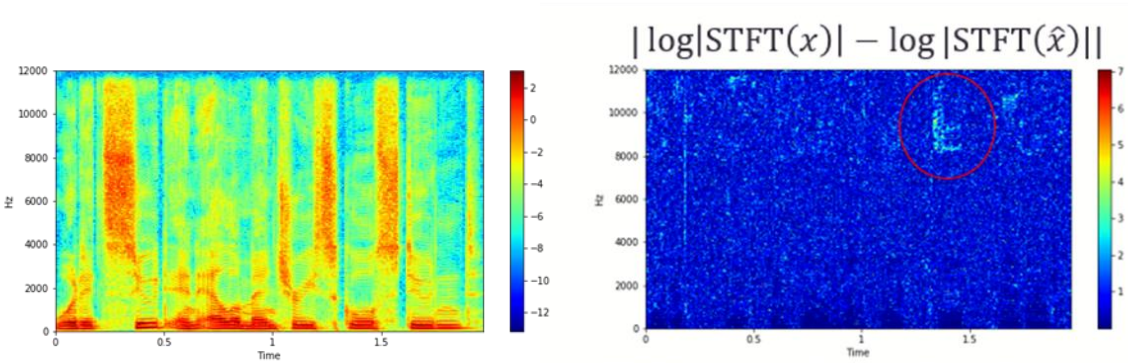
1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,

시간-주파수 축에서 해상도가 다른 여러 개의 Loss 들의 평균이다.

이때, Loss 는 두가지로 구성되는데

하나는 에너지가 큰 구간을 잡아내고

다른 하나는 **에너지가 작은 구간**을 잡아낸다.

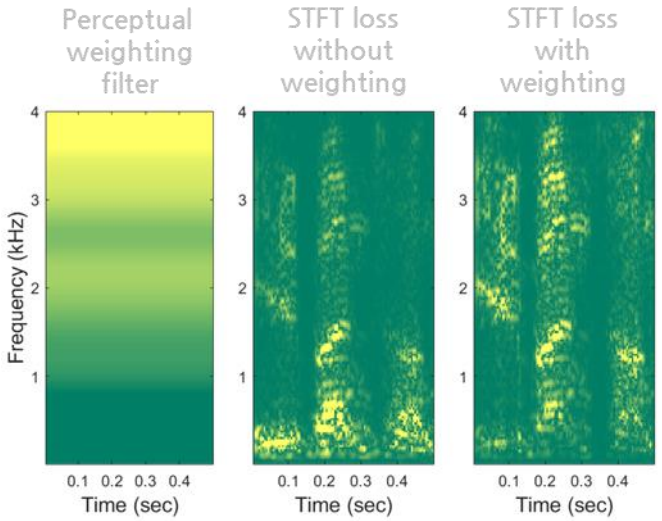


$$L_{\mathrm{mag}}(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{\sum_{t,f} |\log|\mathbf{X}_{t,f}| - \log|\hat{\mathbf{X}}_{t,f}||}{T \cdot N}$$

출처: R. Yamamoto, et al., "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. ICASSP*, 2020.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,
4. STFT Loss 에 Perceptual Weighting Filter 를 적용해서 한번 더 품질을 높인다.



$$L_{\mathrm{sc}}^{w}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sqrt{\sum_{t,f}(\mathbf{W}_{t,f}(|\mathbf{X}_{t,f}| - |\hat{\mathbf{X}}_{t,f}|))^2}}{\sqrt{\sum_{t,f}|\mathbf{X}_{t,f}|^2}}$$

$$L_{\mathrm{mag}}^{w}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sum_{t,f}|\log\mathbf{W}_{t,f}(\log|\mathbf{X}_{t,f}| - \log|\hat{\mathbf{X}}_{t,f}|)|}{T \cdot N}$$

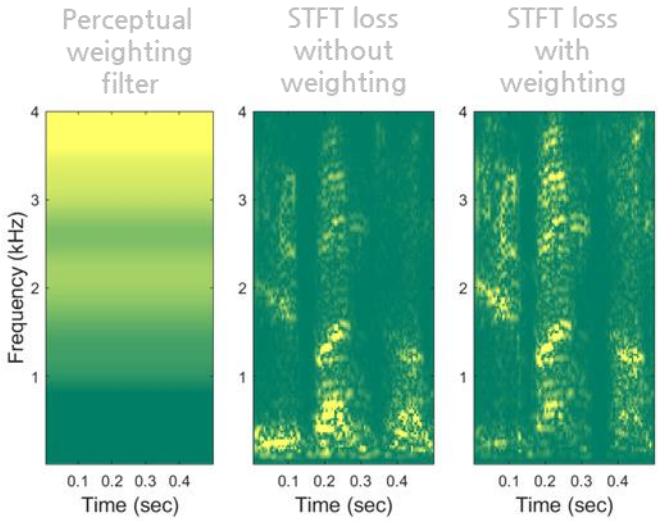
$$\boldsymbol{W}(z) = 1 - \sum_{k=1}^{p}\tilde{\alpha}_k z^{-k}$$

출처: E. Song, et al., “Improved Parallel WaveGAN with perceptually weighted spectrogram loss,” in *Proc. SLT*, 2021.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,
4. STFT Loss 에 Perceptual Weighting Filter 를 적용해서 한번 더 품질을 높인다.

사람에게 청각적으로 더 잘 들리는 **잡음을 제거**하는 역할



$$L_{\mathrm{sc}}^{w}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sqrt{\sum_{t,f}(\mathbf{W}_{t,f}(|\mathbf{X}_{t,f}| - |\hat{\mathbf{X}}_{t,f}|))^2}}{\sqrt{\sum_{t,f}|\mathbf{X}_{t,f}|^2}}$$

$$L_{\mathrm{mag}}^{w}(\boldsymbol{x},\hat{\boldsymbol{x}}) = \frac{\sum_{t,f}|\log(\mathbf{W}_{t,f}(\log|\mathbf{X}_{t,f}| - \log|\hat{\mathbf{X}}_{t,f}|)|}{T \cdot N}$$

$$\boldsymbol{W}(z) = 1 - \sum_{k=1}^{p}\tilde{\alpha}_k z^{-k}$$

출처: E. Song, et al., “Improved Parallel WaveGAN with perceptually weighted spectrogram loss,” in *Proc. SLT*, 2021.

Parallel WaveGAN

1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,
4. STFT Loss 에 Perceptual Weighting Filter 를 적용해서 한번 더 품질을 높인다.

Autoregressive WaveNet		Parallel WaveGAN
합성음 품질이 좋지만 생성 속도가 느리다	VS	학습도 쉽고 생성 속도도 빠르고 합성음 음질도 좋다

300 RT

RT: 1초 음성을 생성할 때 걸리는 시간

Parallel WaveGAN

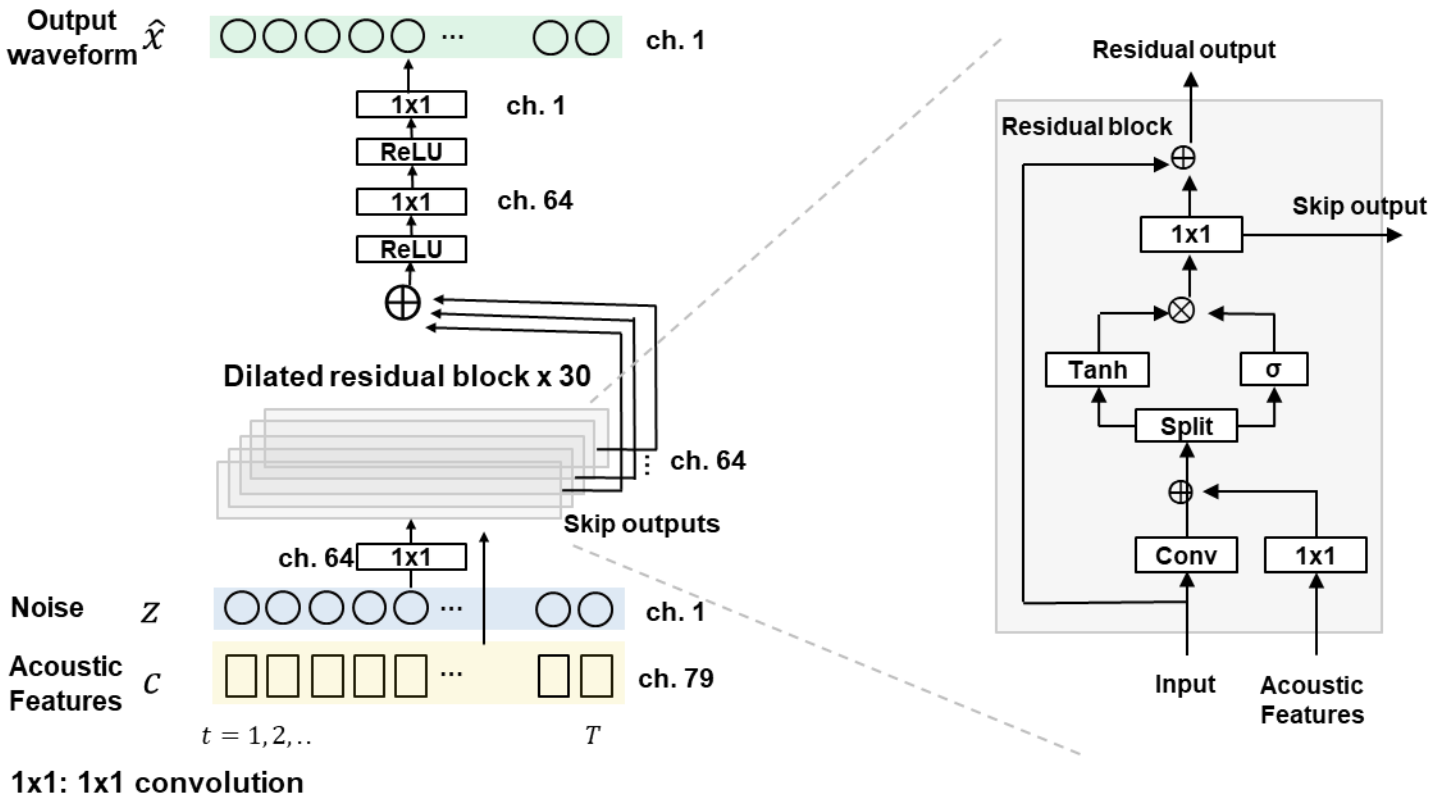
1. Teacher-student 기반의 Probability Distillation 과정을 없애고,
2. Adversarial Training 으로 합성음 품질을 높이고,
3. Multi-resolution STFT Loss 로 합성음 품질을 더 높이고,
4. STFT Loss 에 Perceptual Weighting Filter 를 적용해서 한번 더 품질을 높인다.

Autoregressive WaveNet		Parallel WaveGAN
합성음 품질이 좋지만 생성 속도가 느리다	VS	학습도 쉽고 생성 속도도 빠르고 합성음 음질도 좋다
		
		0.02 RT

RT: 1초 음성을 생성할 때 걸리는 시간

Parallel WaveGAN

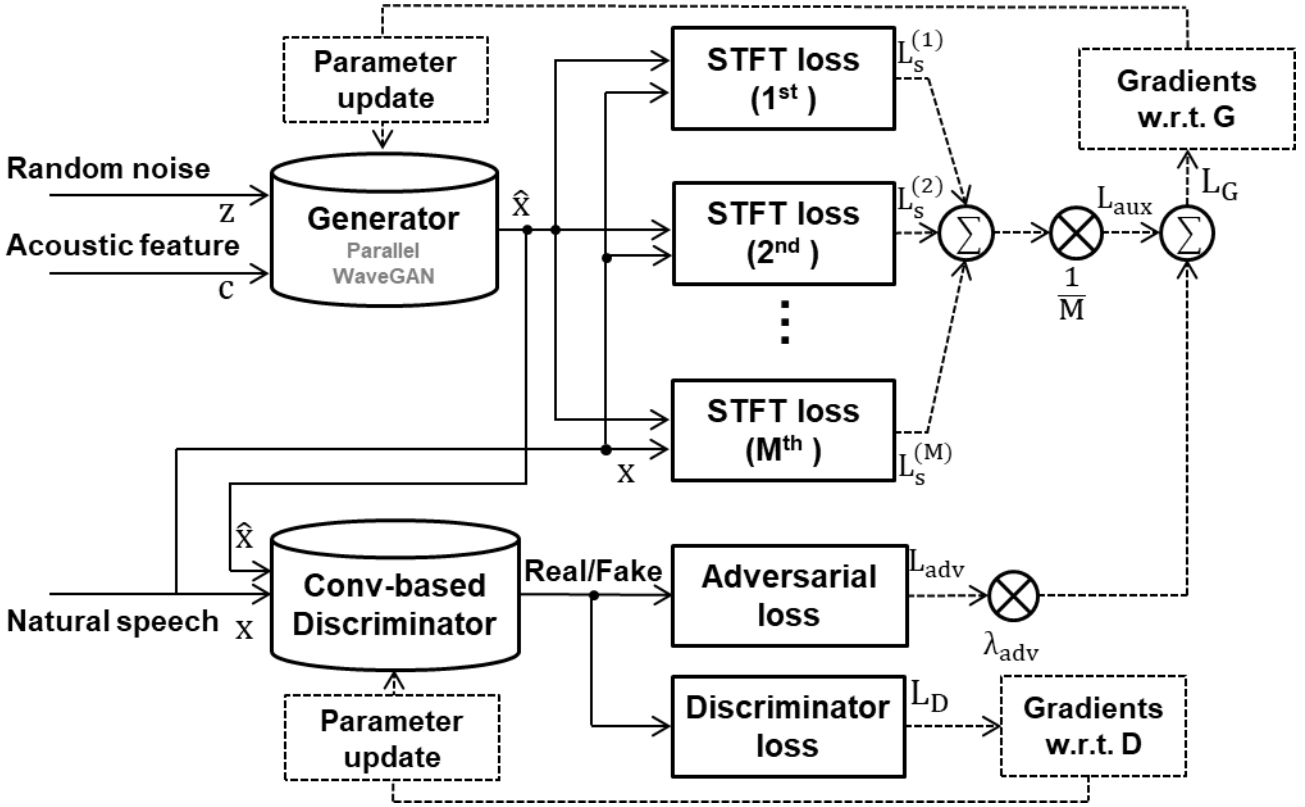
모델 구조



출처: R. Yamamoto, et al., "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. ICASSP*, 2020.

Parallel WaveGAN

학습 방법



출처: R. Yamamoto, et al., "Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram," in *Proc. ICASSP*, 2020.

Summary

Autoregressive 생성 방법과 Non-autoregressive 생성 방법을 꼭 기억해 주세요!

Autoregressive vocoder

- Sample-by-sample generation
  - $p(\mathbf{x}|\mathbf{h}) = \prod_{t=1}^{T} p(x_t|x_1, \cdots, x_{t-1}, \mathbf{h})$
  - $\mathbf{h}$: Conditional acoustic parameter

Non-autoregressive vocoder

- Parallel generation
  - $p(\mathbf{x}|\mathbf{h}) = \prod_{t=1}^{T} p(x_t|z_1, \cdots, z_{t-1}, \mathbf{h})$
  - z_i: Random variable
  - $\mathbf{h}$: Conditional acoustic parameter

Teacher-student distillation

- Parallel WaveNet, ClariNet

GAN-based approaches

- Parallel WaveGAN
- MelGAN, VocGAN, Hi-Fi GAN

CLOVA

Q / A

Acoustic model

Statistical parametric speech synthesis

Recall

Acoustic model 은 **Text** 로부터 **Acoustic Parameter** 를 추정하는 역할을 합니다.

Parametric LPC vocoder

WaveNet vocoder

Tacotron 2

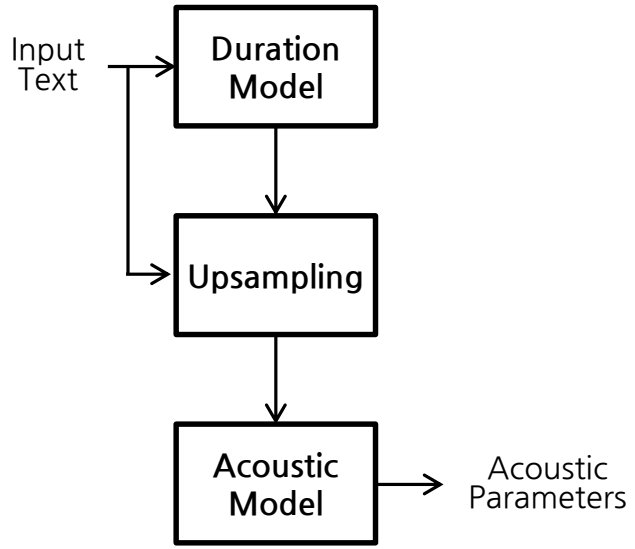
출처: J. Shen, et. al., “Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions,” in *Proc. ICASSP*, 2018.

Overview

Acoustic model 은 **Text** 로부터 **Acoustic Parameter** 를 추정하는 역할을 합니다.

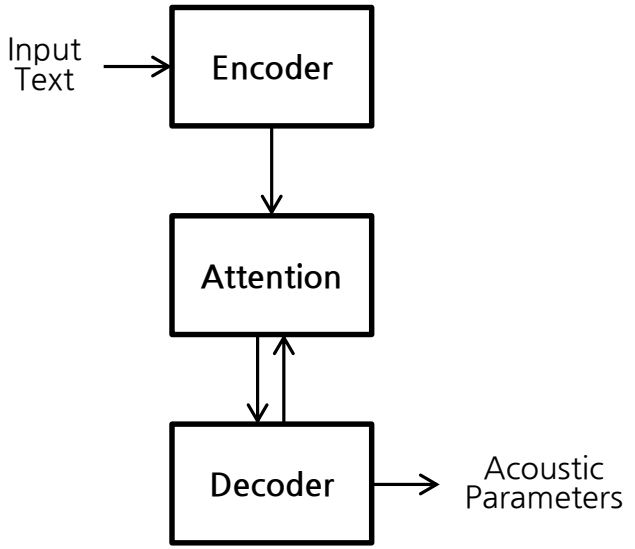
Statistical parametric speech synthesis

- Simple deep learning model (FF+LSTM)

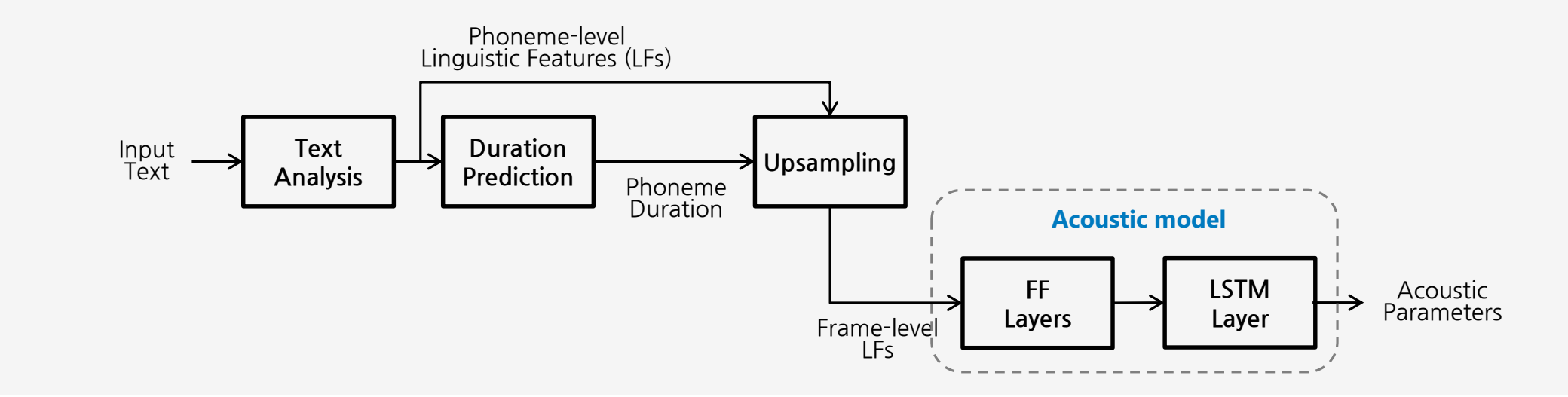


End-to-end speech synthesis

- Seq2seq model

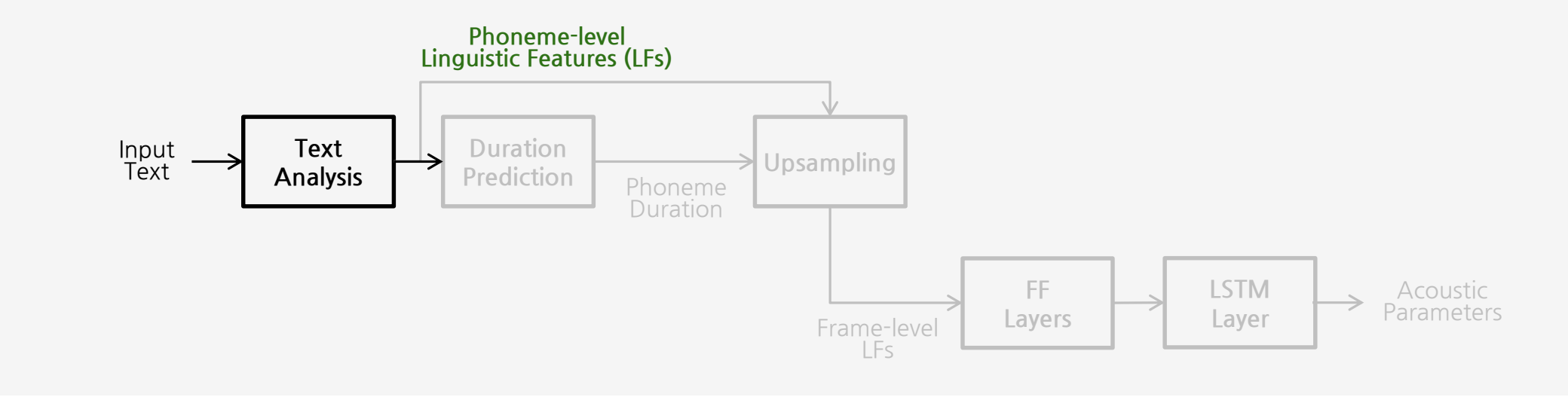


Statistical parametric speech synthesis (SPSS)

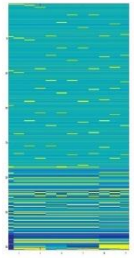


Statistical parametric speech synthesis (SPSS)

Text analyzer: Generates phoneme-level linguistic features (Phoneme: 음운론상의 최소 단위)



"안녕하세요"



Phoneme ID
Break
Punctuation
Position
.
.

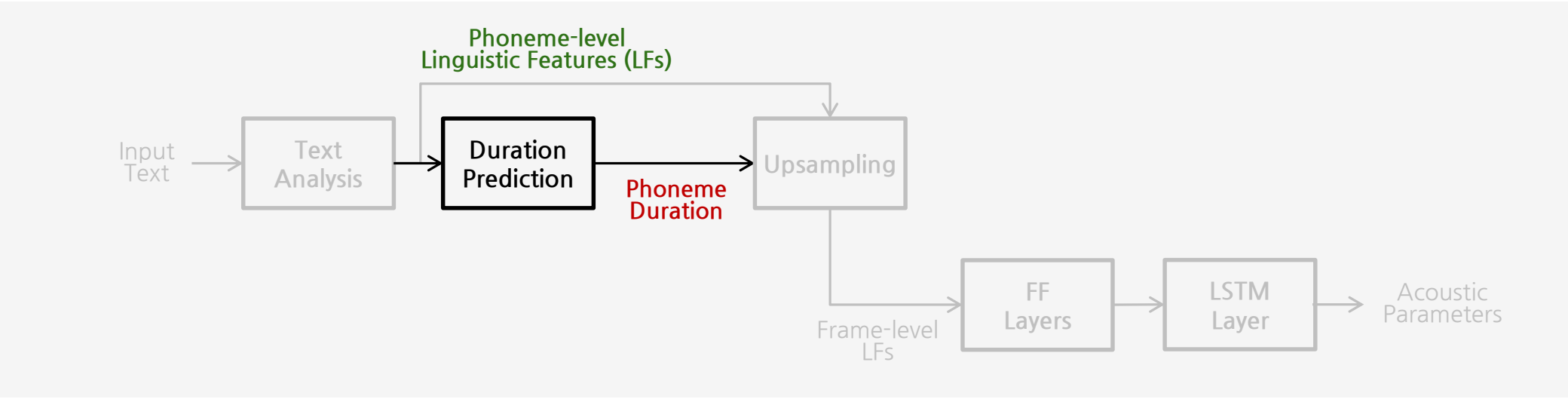
a-N| n-jv-O| h-a| s-e| jo

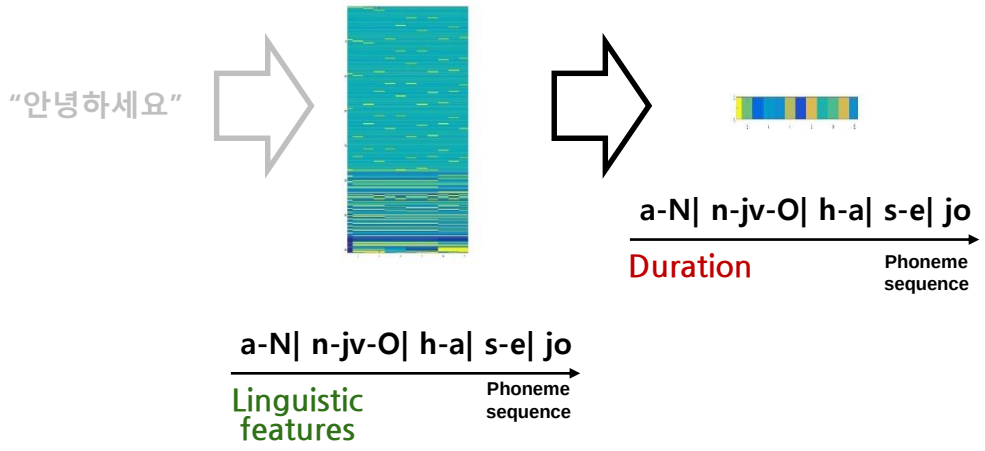
Linguistic features

Phoneme sequence

Statistical parametric speech synthesis (SPSS)

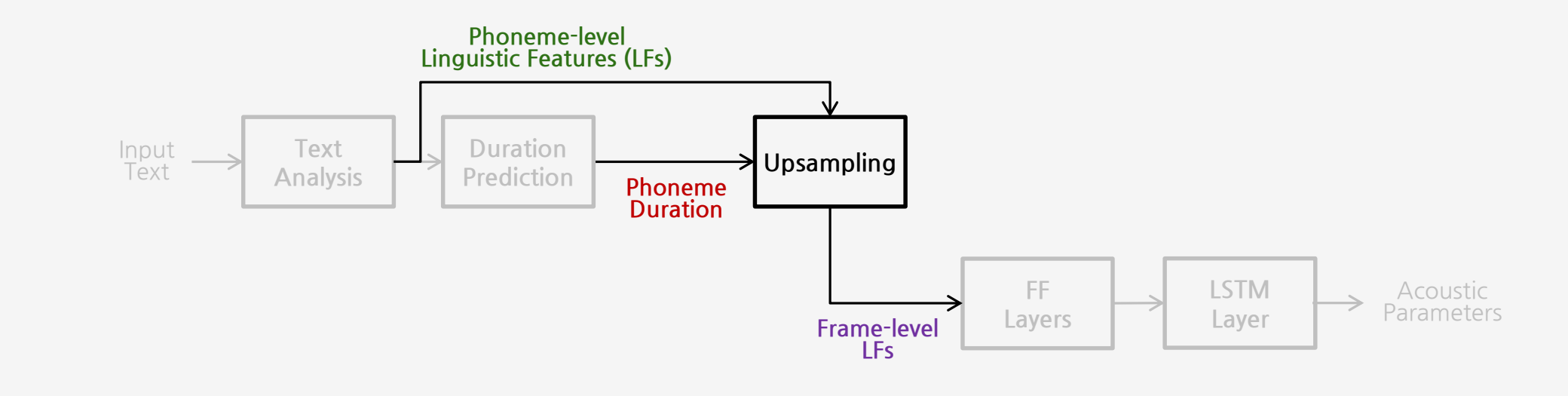
Duration model: Predicts phoneme duration

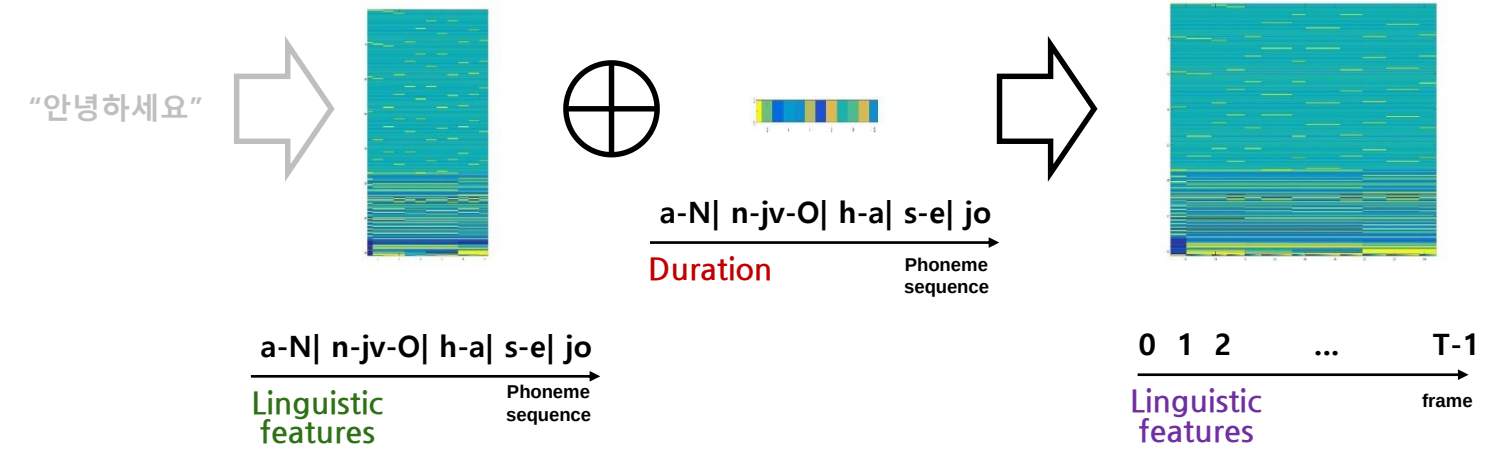




Statistical parametric speech synthesis (SPSS)

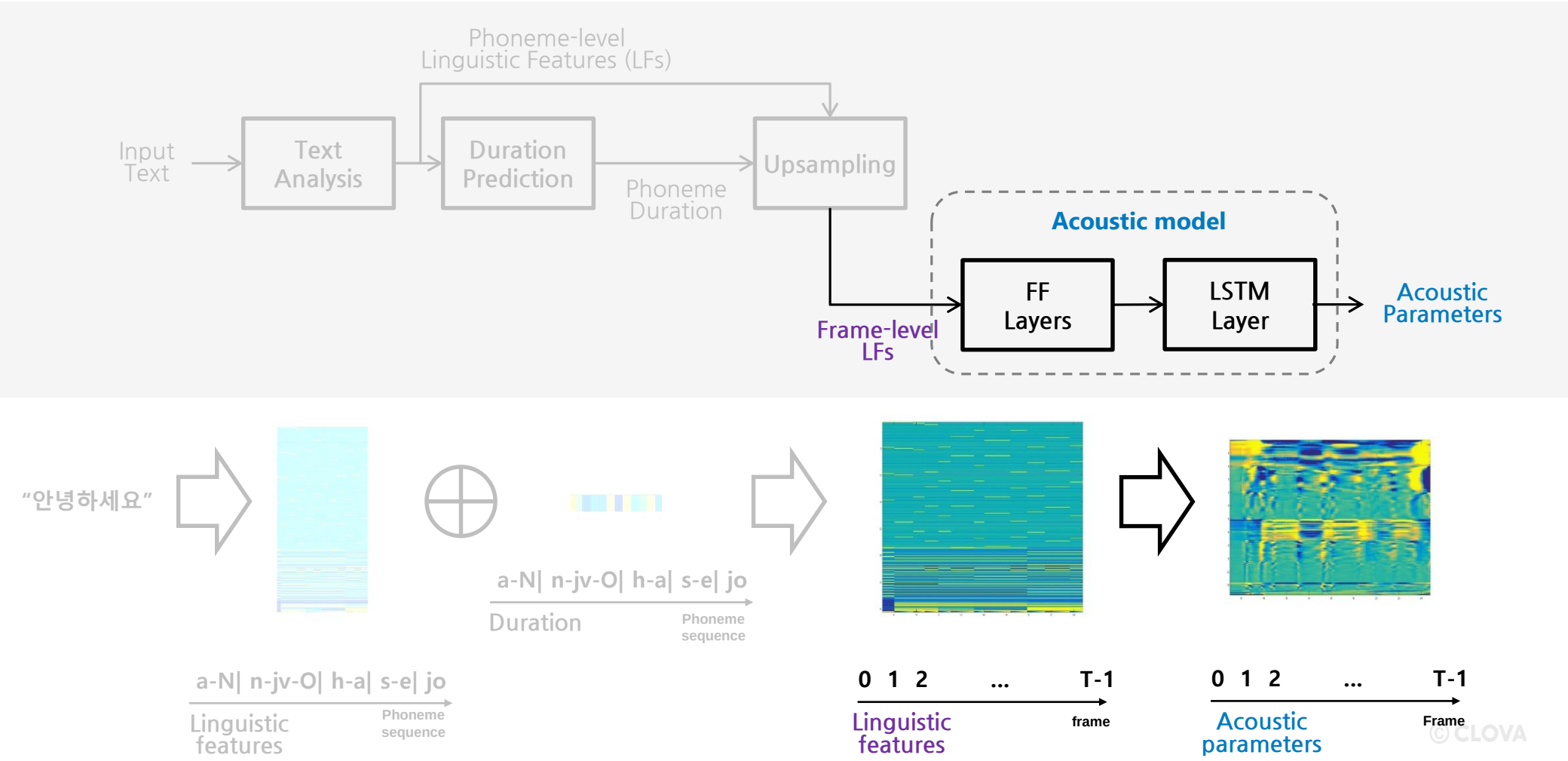
Linguistic upsampler: Generates frame-level linguistic features





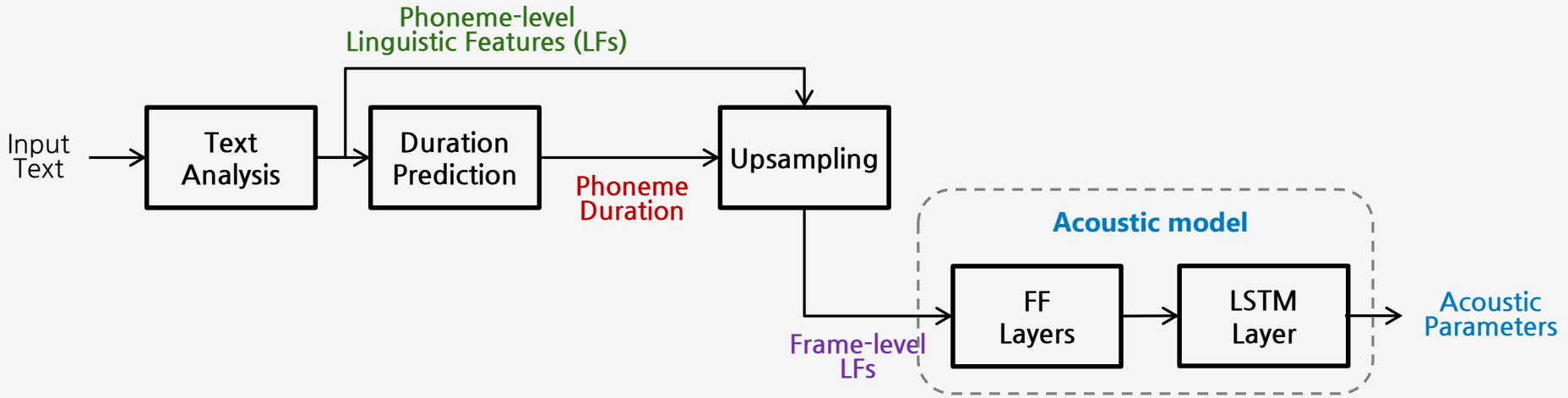
Statistical parametric speech synthesis (SPSS)

Acoustic model: Predicts frame-level acoustic parameters



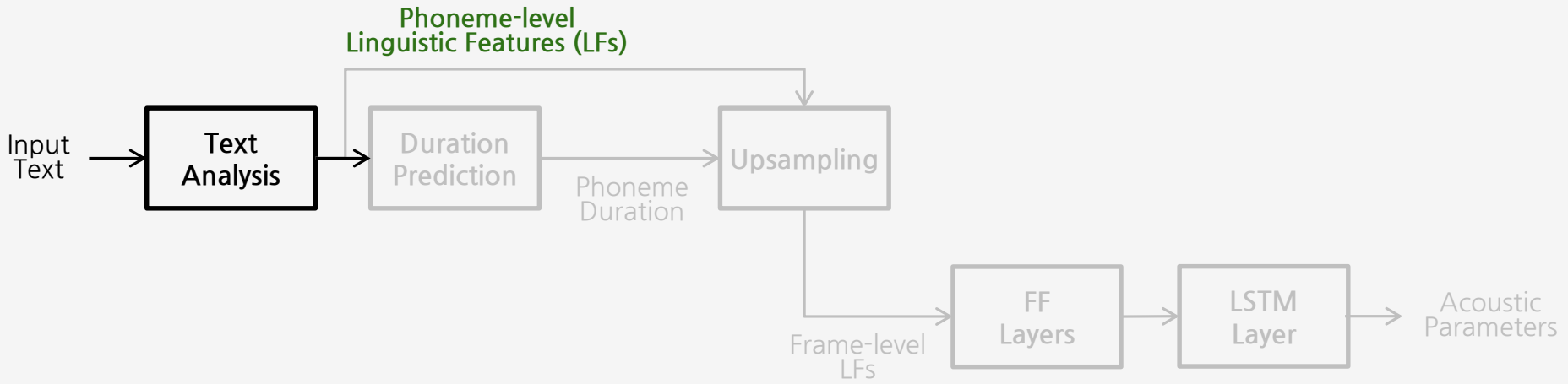
Statistical parametric speech synthesis (SPSS)

Frame-level Linguistic Feature 를 만드는게 너무 어려워요 ...



Statistical parametric speech synthesis (SPSS)

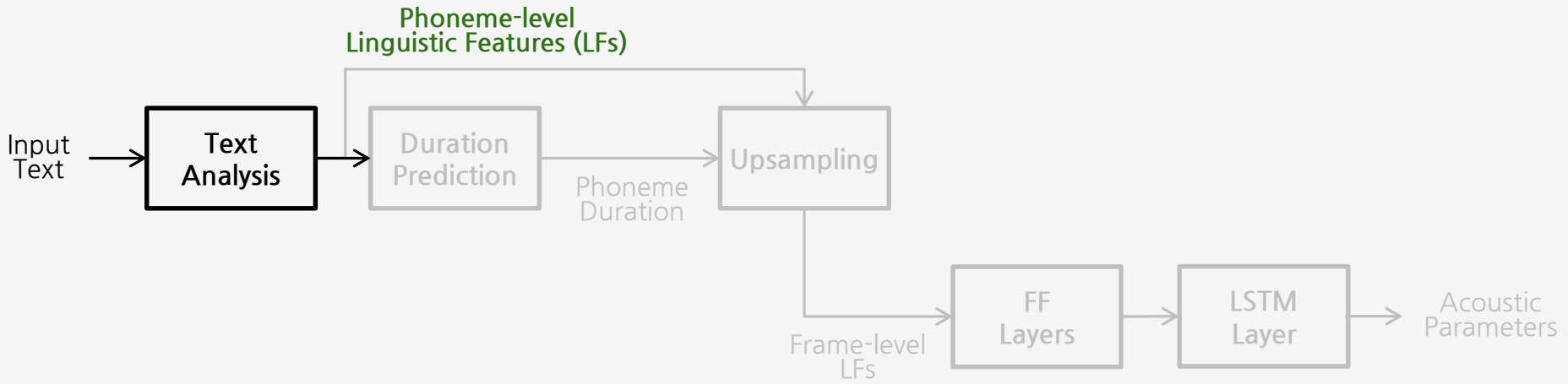
Frame-level Linguistic Feature 를 만드는게 너무 어려워요 ...



약 **500m** 앞 남동 IC에서 인천광역시청, 인천지방경찰청, 남동공단 방면 오른쪽 고속도로 출구입니다.

Statistical parametric speech synthesis (SPSS)

Frame-level Linguistic Feature 를 만드는게 너무 어려워요 ...



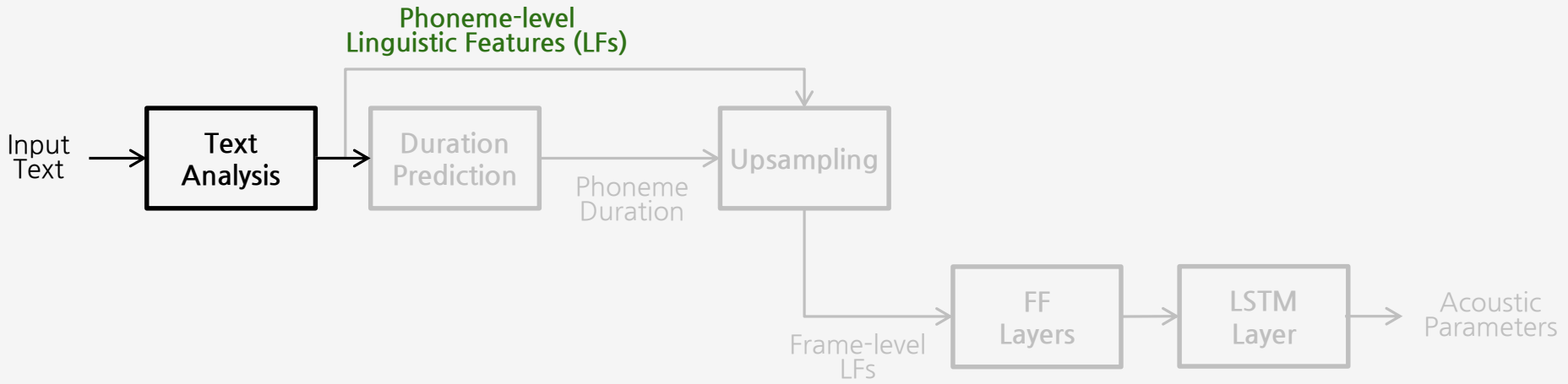
약 **500m** 앞 남동 IC에서 인천광역시청, 인천지방경찰청, 남동공단 방면 오른쪽 고속도로 출구입니다.

오백미터? 오백미리? 다섯영영엠?

"트와이쓰의 노크노크를 재생합니다"...

Statistical parametric speech synthesis (SPSS)

Frame-level Linguistic Feature 를 만드는게 너무 어려워요 ...



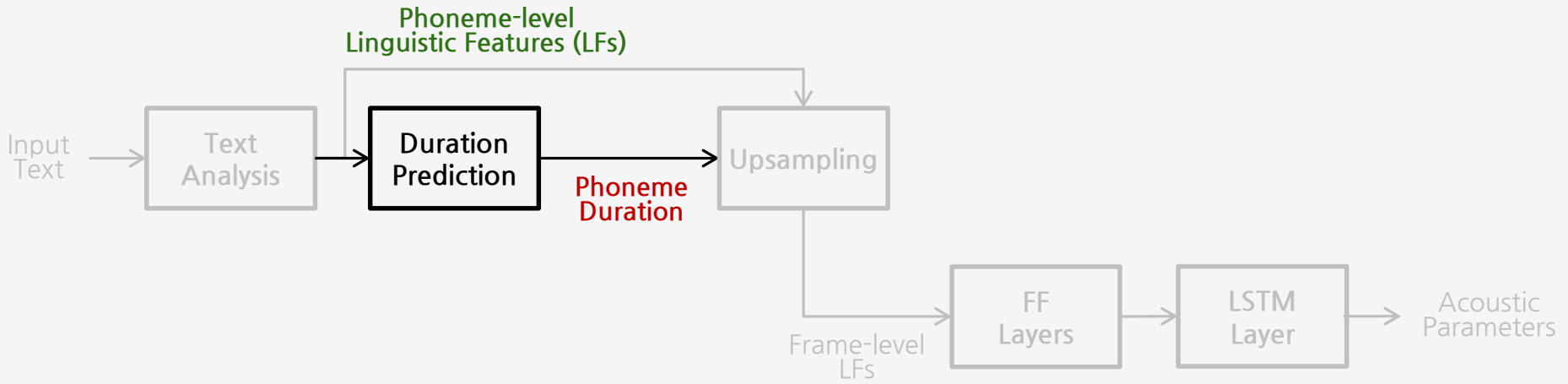
약 **500m** 앞 남동 IC에서 인천광역시청, 인천지방경찰청, 남동공단 방면 오른쪽 고속도로 출구입니다.

Text normalization, grapheme-to-phoneme, break estimator 등
고도화된 **NLP front-end** 가 필요합니다.

약 오뱅미터 앞 | 남동아이씨에서 | 인천광역씨청 | 인천지방경찰청 | 남동공단방면 | 오른쪽 고속또로 출구임니다

Statistical parametric speech synthesis (SPSS)

Frame-level Linguistic Feature 를 만드는게 너무 어려워요 ...



모든 [스크립트 - 녹음] DB 에서

Phoneme-level Segmentation 데이터가 있어야

Duration Model 을 학습할 수 있습니다.

"일차"

0.000~0.941초: 묵음

0.941~1.034초: /ㅣ/

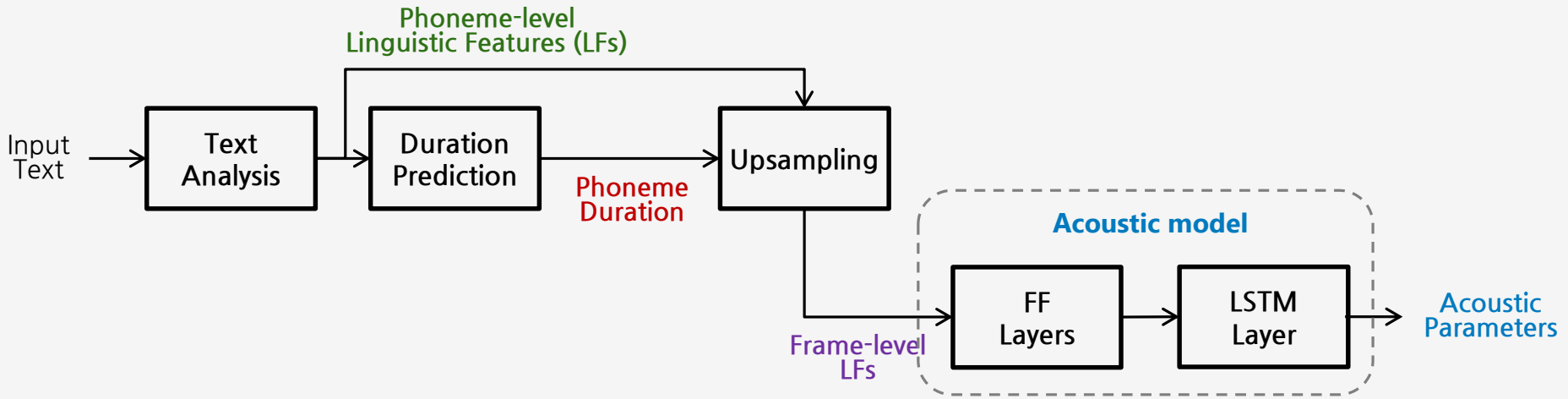
1.034~1.095초: /ㄹ/

1.095~1.191초: /ㅊ/

1.191~1.263초: /ㅏ/

Statistical parametric speech synthesis (SPSS)

Frame-level Linguistic Feature 를 만드는게 너무 어려워요 ...



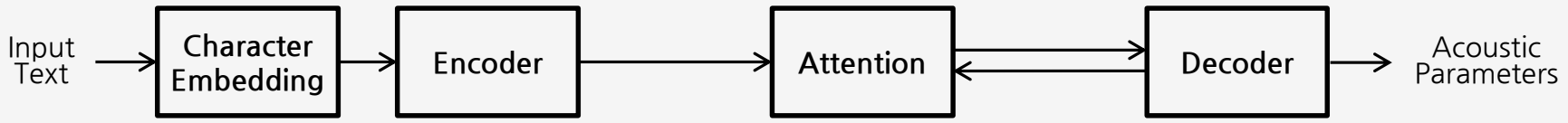
Feature engineering 을 최소화 할 수 없을까 ..?

Acoustic model

End-to-end speech synthesis

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.



End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

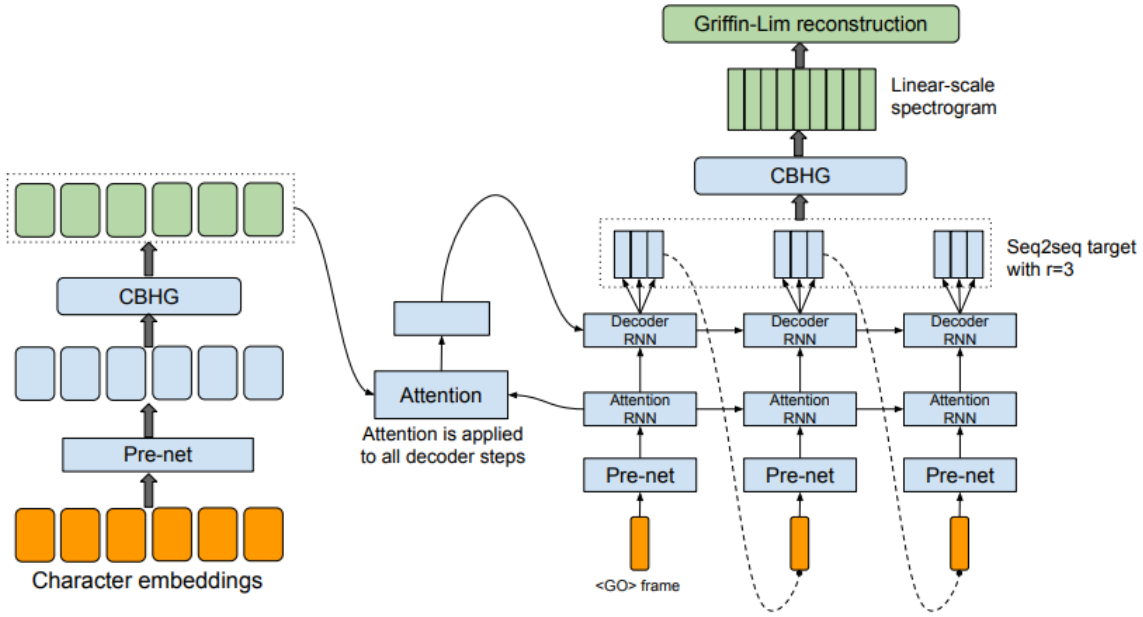


Figure 1: *Model architecture. The model takes characters as input and outputs the corresponding raw spectrogram, which is then fed to the Griffin-Lim reconstruction algorithm to synthesize speech.*

Tacotron

출처: Y. Wu, et al., "Tacotron: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

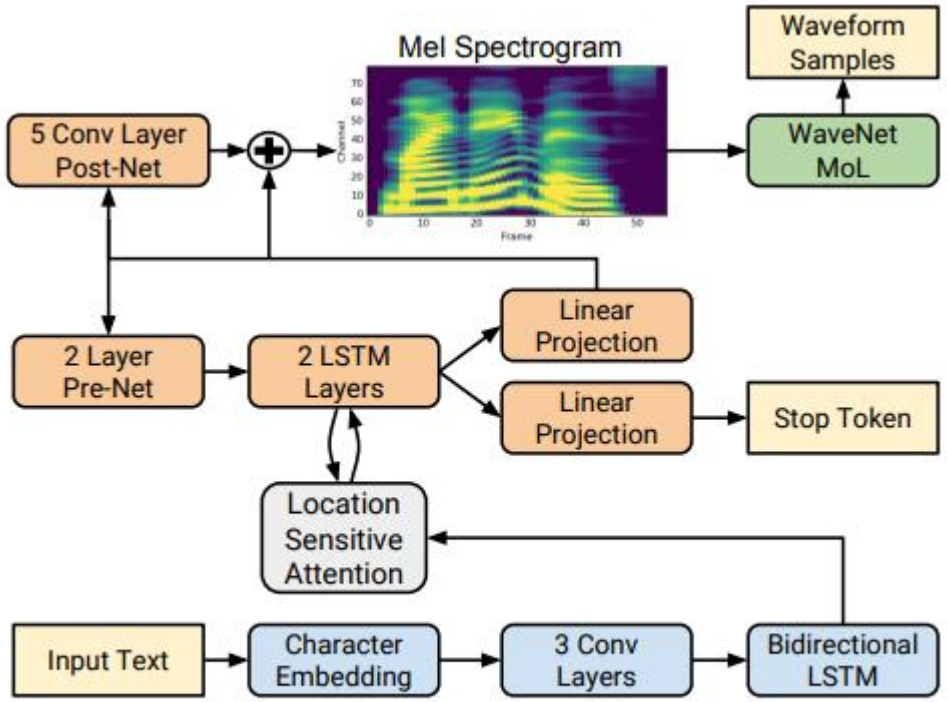


Fig. 1. Block diagram of the Tacotron 2 system architecture.

Tacotron 2

출처: J. Shen, et. al., "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *Proc. ICASSP*, 2018.

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

System	MOS
Parametric	3.492 ± 0.096
Tacotron (Griffin-Lim)	4.001 ± 0.087
Concatenative	4.166 ± 0.091
WaveNet (Linguistic)	4.341 ± 0.051
Ground truth	4.582 ± 0.053
Tacotron 2 (this paper)	$\mathbf{4.526 \pm 0.066}$

Table 1. Mean Opinion Score (MOS) evaluations with 95% confidence intervals computed from the t-distribution for various systems.

Tacotron 2

출처: J. Shen, et. al., "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," in *Proc. ICASSP*, 2018.

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

Tacotron 2 learns pronunciations based on phrase semantics.

(Note how Tacotron 2 pronounces "read" in the first two phrases.)

"He has read the whole thing."

0:01 / 0:01

"He reads books."

0:00 / 0:00

출처: https://google.github.io/tacotron/publications/tacotron2/

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

Tacotron 2 learns pronunciations based on phrase semantics.

(Note how Tacotron 2 pronounces "read" in the first two phrases.)

"He has read the whole thing."

"He reads books."

출처: https://google.github.io/tacotron/publications/tacotron2/

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

Tacotron 2 is somewhat robust to spelling errors.

"Thisss isrealy awhsome."

0:01 / 0:01

Tacotron 2 is sensitive to punctuation.

(Note how the comma in the first phrase changes prosody.)

"This is your personal assistant, Google Home."

0:02 / 0:02

"This is your personal assistant Google Home."

0:02 / 0:02

출처: https://google.github.io/tacotron/publications/tacotron2/

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

Tacotron 2 is somewhat robust to spelling errors.

"Thisss isrealy awhsome."

0:01 / 0:01

Tacotron 2 is sensitive to punctuation.

(Note how the comma in the first phrase changes prosody.)

"This is your personal assistant, Google Home."

0:02 / 0:02

"This is your personal assistant Google Home."

0:02 / 0:02

출처: https://google.github.io/tacotron/publications/tacotron2/

End-to-end speech synthesis

(Text) **Encoder** 와 (Acoustic Parameter) **Decoder** 를 만들고, **Attention** 으로 Alignment 를 잡아주면 됩니다.

Tacotron 2 is somewhat robust to spelling errors.

"Thisss isrealy awhsome."

0:01 / 0:01

Tacotron 2 is sensitive to punctuation.

(Note how the comma in the first phrase changes prosody.)

"This is your personal assistant, Google Home."

0:02 / 0:02

"This is your personal assistant Google Home."

0:02 / 0:02

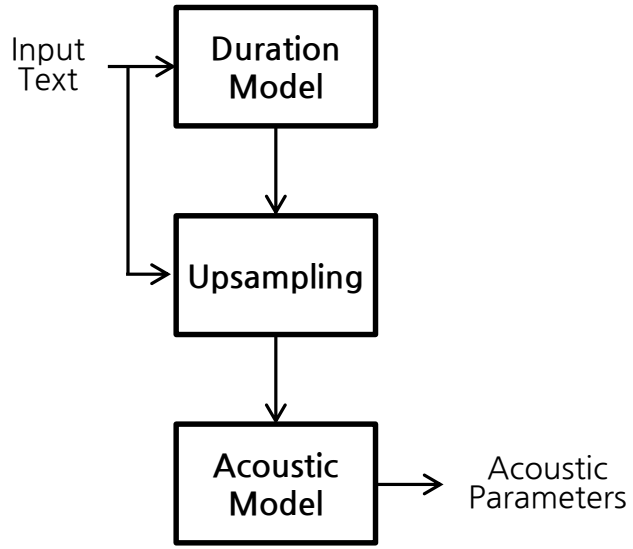
출처: https://google.github.io/tacotron/publications/tacotron2/

Summary

End-to-end Tacotron 모델을 꼭 기억해주세요

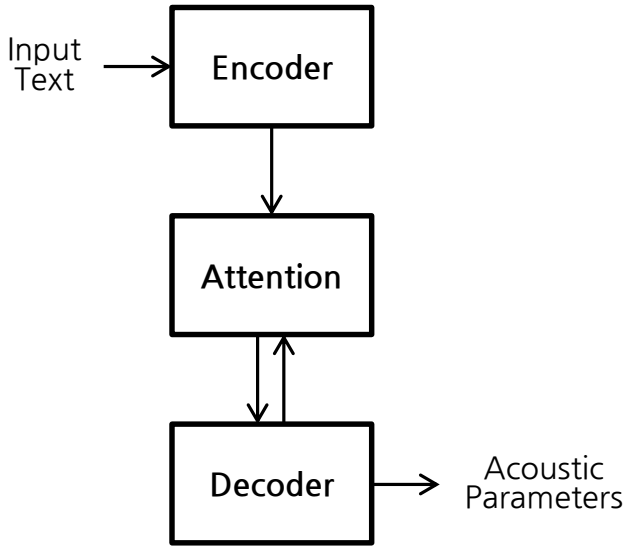
Statistical parametric speech synthesis

- Simple deep learning model (FF+LSTM)



End-to-end speech synthesis

- Seq2seq model



Summary

Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.

DNN TTS = Acoustic model + Vocoder

Summary

Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.

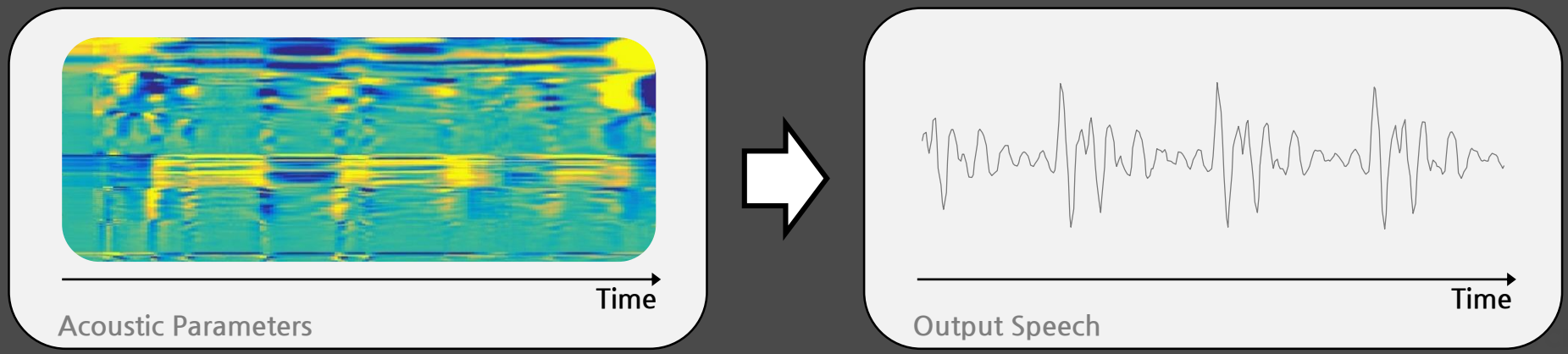
톤의 높낮이, 음색, 어조, 강세 등

텍스트에서 **Acoustic Parameter** 를 추정

Summary

Text-to-speech (TTS) 란 기계가 사람처럼 **텍스트를 읽어주는** 기술입니다.

Acoustic Parameter 에서 음성 신호를 추정



CLOVA

Q / A

CLOVA

Text-to-speech

CLOVA 음성 합성 서비스

Unit-selection TTS

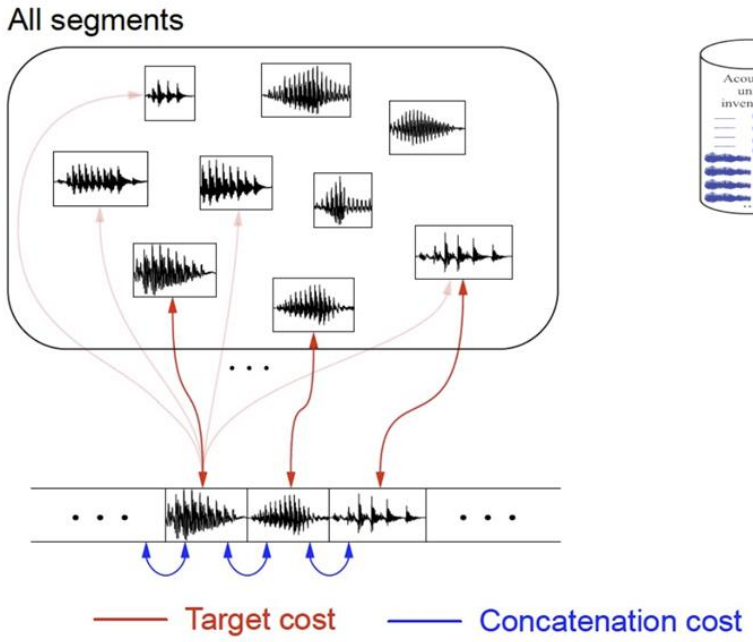
문맥에 맞는 유닛을 붙여서 음성을 만들자

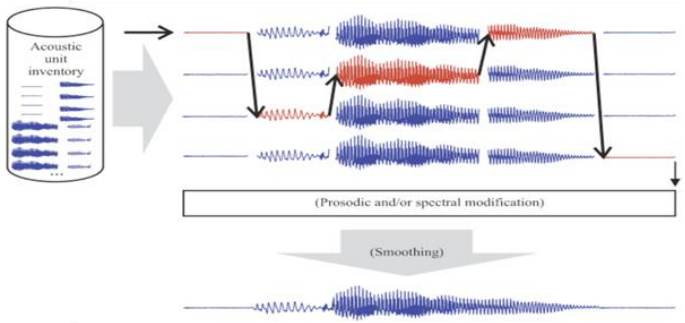
Input
Text

Unit-selection TTS

문맥에 맞는 유닛을 붙여서 음성을 만들자

Output Speech



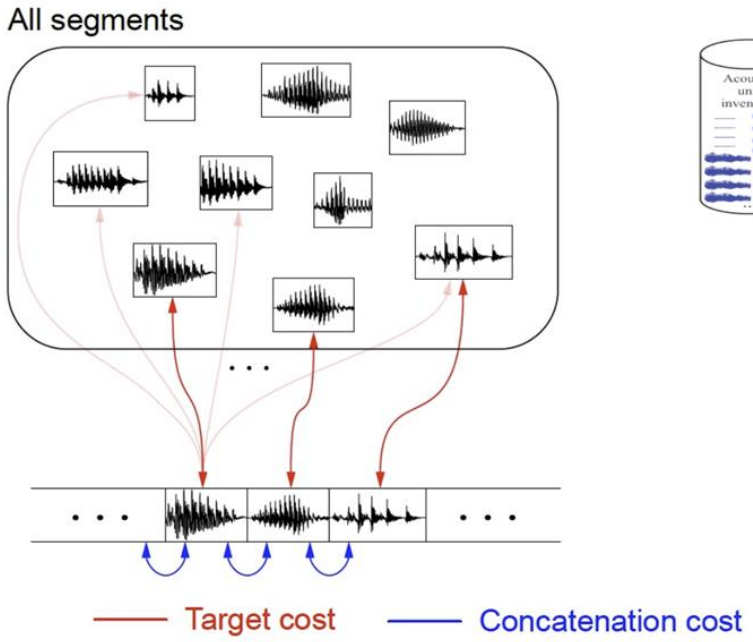


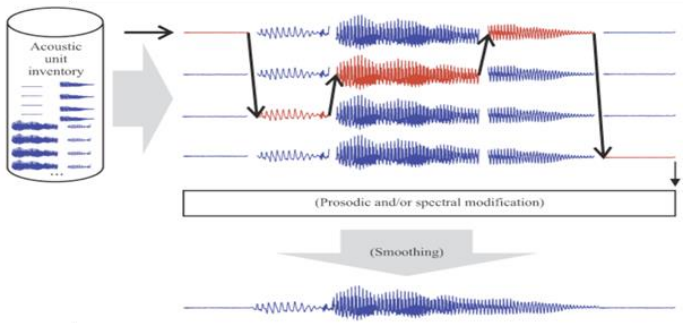
$$C(t_1^n, u_1^n) = \sum_{i=1}^{n} C^{target}(t_i, u_i) + \sum_{i=2}^{n} C^{join}(u_{i-1}, u_i)$$

$$\hat{u}_1^n = \underset{u_1,\ldots,u_n}{\operatorname{argmin}} C(t_1^n, u_1^n)$$

그림 출처: T. Keiichi, and H. Zen. "Fundamentals and recent advances in HMM-based speech synthesis." *Tutorial of INTERSPEECH,* 2009.
기술 출처: A. Hunt and A. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” in *Proc. ICASSP*, 1996.

문맥에 맞는 유닛을 붙여서 음성을 만들자





$$C(t_1^n, u_1^n) = \sum_{i=1}^{n} C^{target}(t_i, u_i) + \sum_{i=2}^{n} C^{join}(u_{i-1}, u_i)$$

$$\hat{u}_1^n = \underset{u_1,\ldots,u_n}{\operatorname{argmin}} C(t_1^n, u_1^n)$$

그림 출처: T. Keiichi, and H. Zen. "Fundamentals and recent advances in HMM-based speech synthesis." *Tutorial of INTERSPEECH,* 2009.
기술 출처: A. Hunt and A. Black, “Unit selection in a concatenative speech synthesis system using a large speech database,” in *Proc. ICASSP* 1996.

지금까지도 모든 TTS 서비스에 사용되는 기술

Whale Browser

papago

Navigation

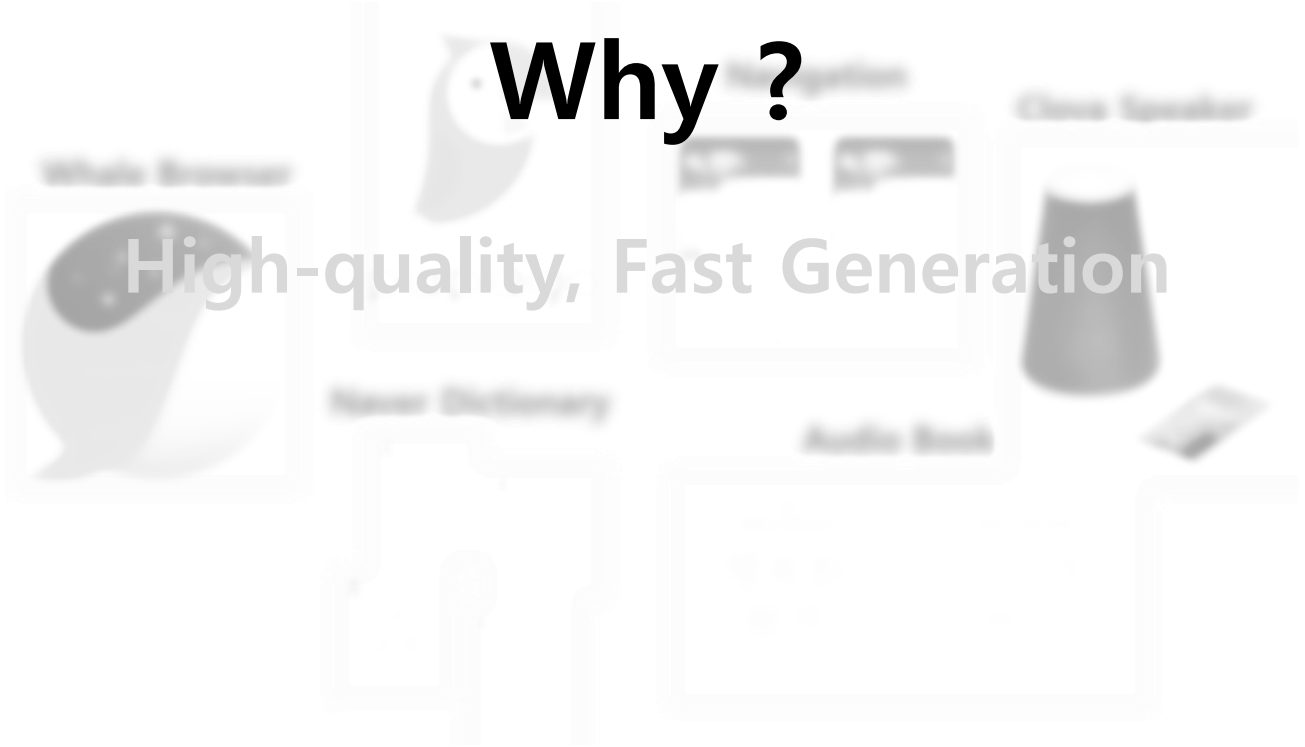
Clova Speaker

Naver Dictionary

Audio Book

네이버 TTS 서비스의 90%에 사용

Why ?



High-quality, Fast Generation

네이버 TTS 서비스의 90%에 사용

High-quality, Fast Generation

네이버 TTS 서비스의 90%에 사용

100hrs UTS 방식의 녹음 필요 시간

1.5yrs 약 1.5년여의 개발 기간

엔진 만들기가 쉽지 않습니다..

HDTS

(High-quality DNN Text-to-Speech)

음성 접합 DB 확보 Concatenative

자연스러움

Unit-Selection 고된 녹음 대용량

실시간 합성

UTS

Generative 오랜 생성 시간

Adaptation

WaveNet 로봇 같은 소리 딥러닝

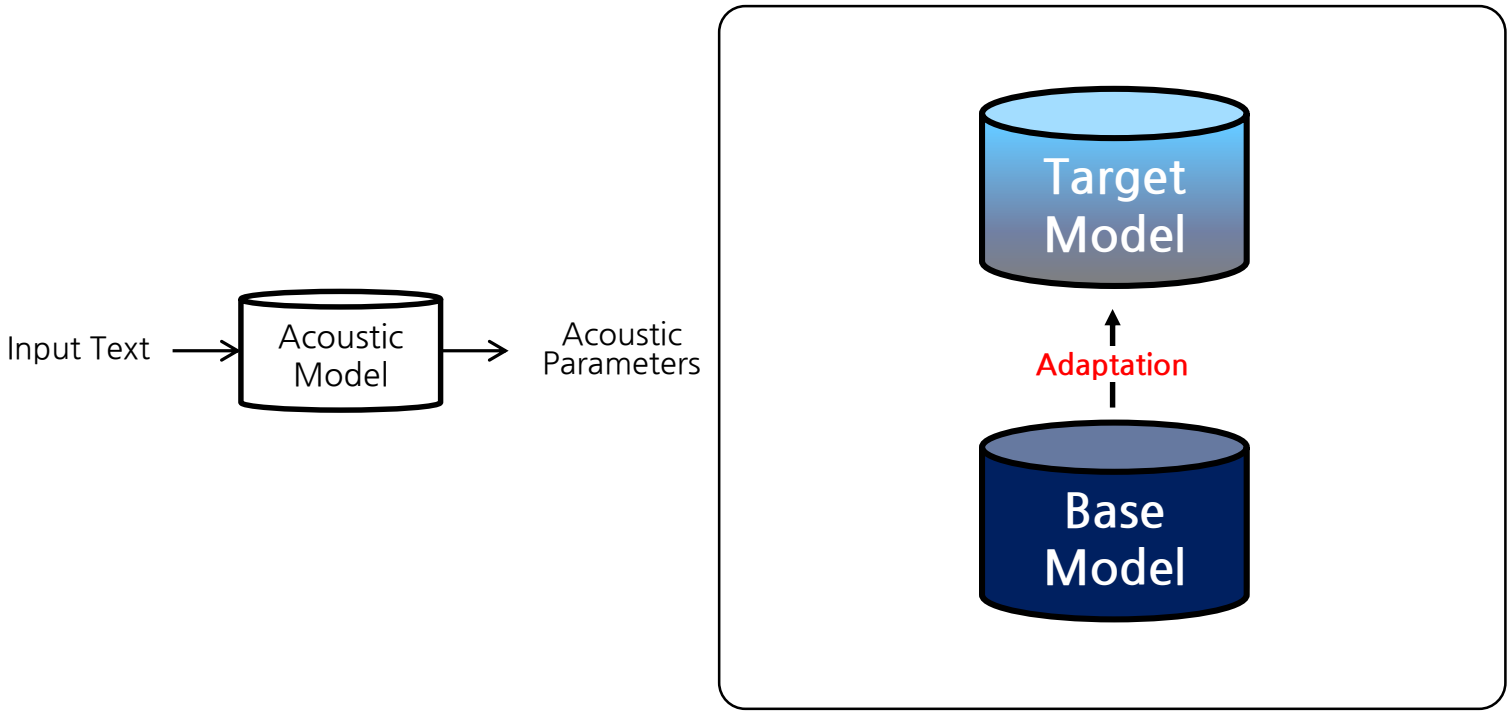
저비용

음성 생성

DTS

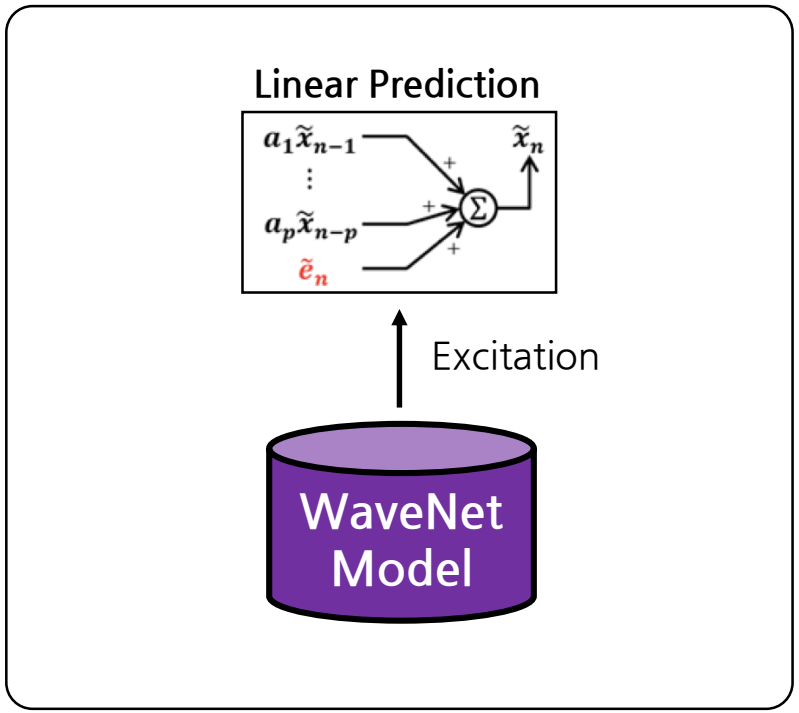
HDTS

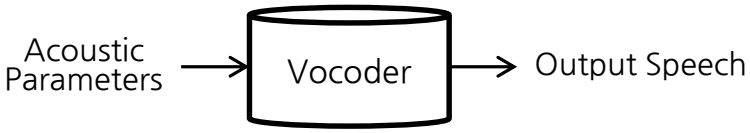
Acoustic Model + Vocoding Model



1. Speaker Adaptation

100hrs → 4hrs (녹음 데이터 4% 이하로 감소)





2. LP-WaveNet Synthesis
합성음 품질 2배 향상 (MOS 2.3 → 4.5)

Speaker Adaptation

DB 녹음 4시간

LP-WaveNet 보코더

MOS 4.5

UTS + DTS 구조 결합

CPU + 0.01초

개발 비용 95% 감소

개발 기간 90% 감소

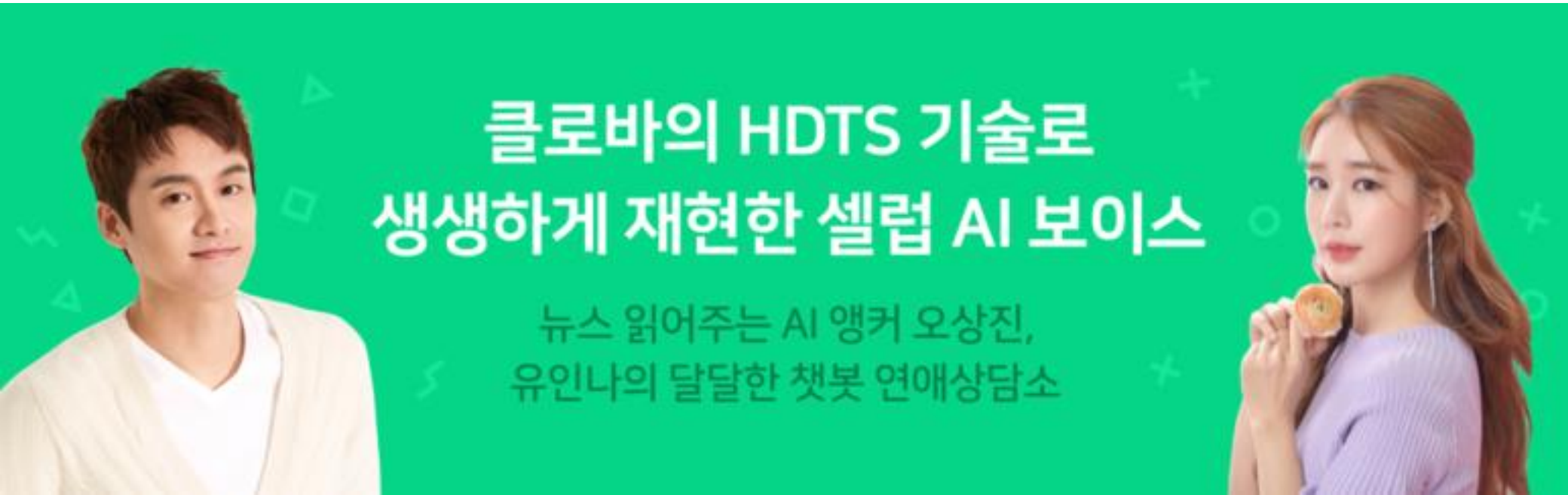
Line Conference
2주의 개발 기간, 음성 녹음 4시간 (2018.06)

+

‘유인나’ Voice
클로바 스피커 기본 목소리 적용 (2018.11)

Clova

‘오상진’ Voice
네이버 뉴스 본문 듣기 적용 (2020.05)



Clova

‘오상진’ Voice

네이버 뉴스 본문

Clova

‘오상진’ Voice

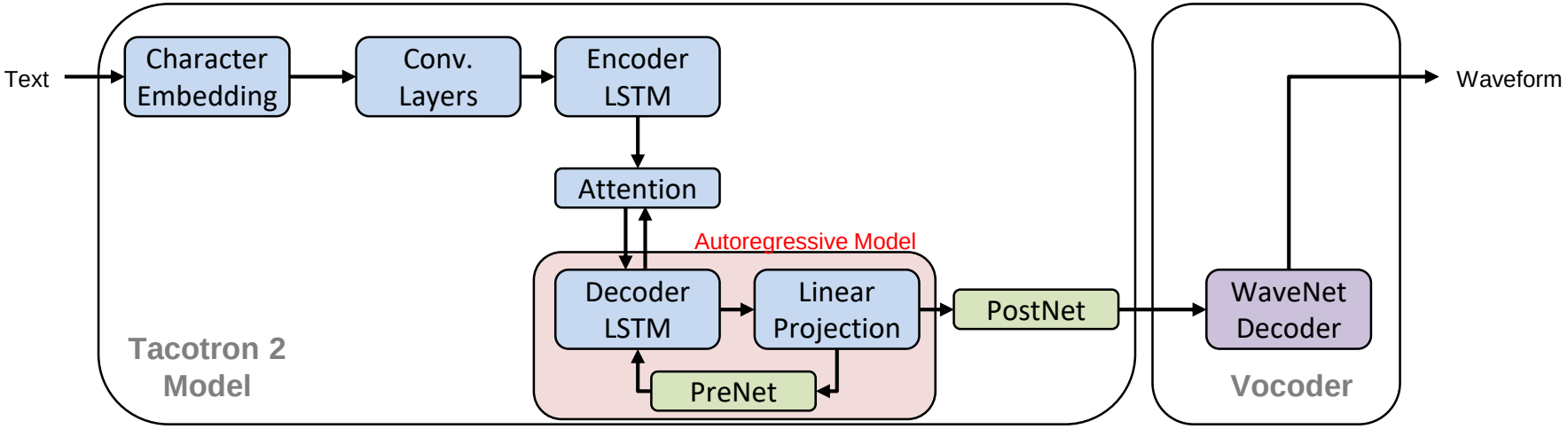
네이버 뉴스 본문

‘유인나’ Voice
네이버 지도 내비게이션 안내 (2020. 09)

Text-to-speech

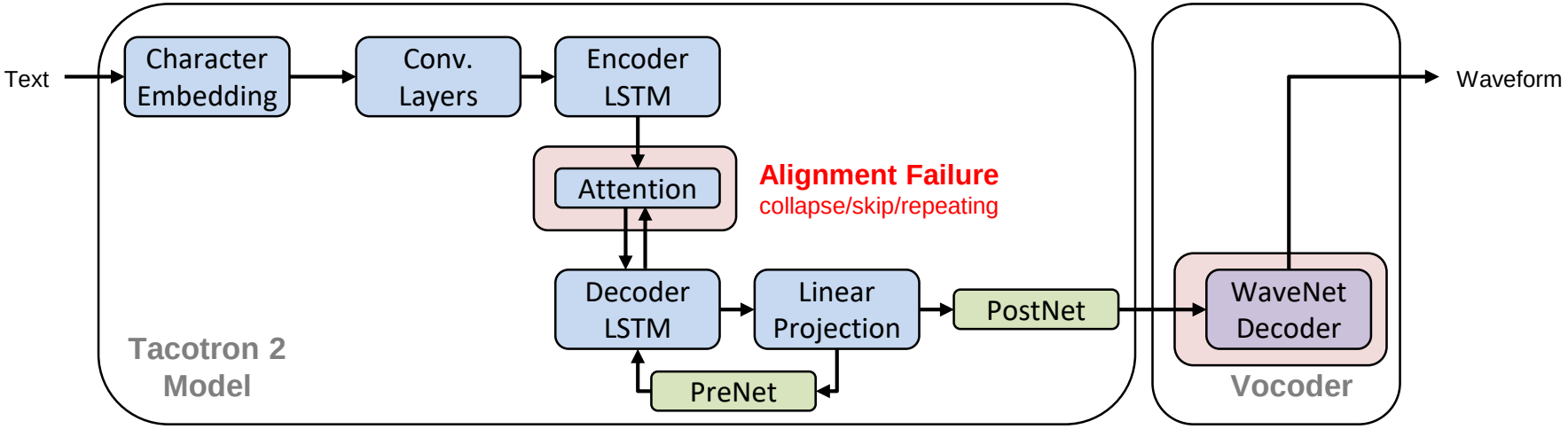
CLOVA HDTS

Recall



Seq2seq model with attention	Phoneme Duration 없어도됨
Autoregressive acoustic model	Acoustic Parameter 추정 정확도가 높아짐
WaveNet vocoder	CNN 모델이 음성 생성해줌

Recall



Seq2seq model with attention

Autoregressive acoustic model

WaveNet vocoder

Phoneme Duration 없어도됨

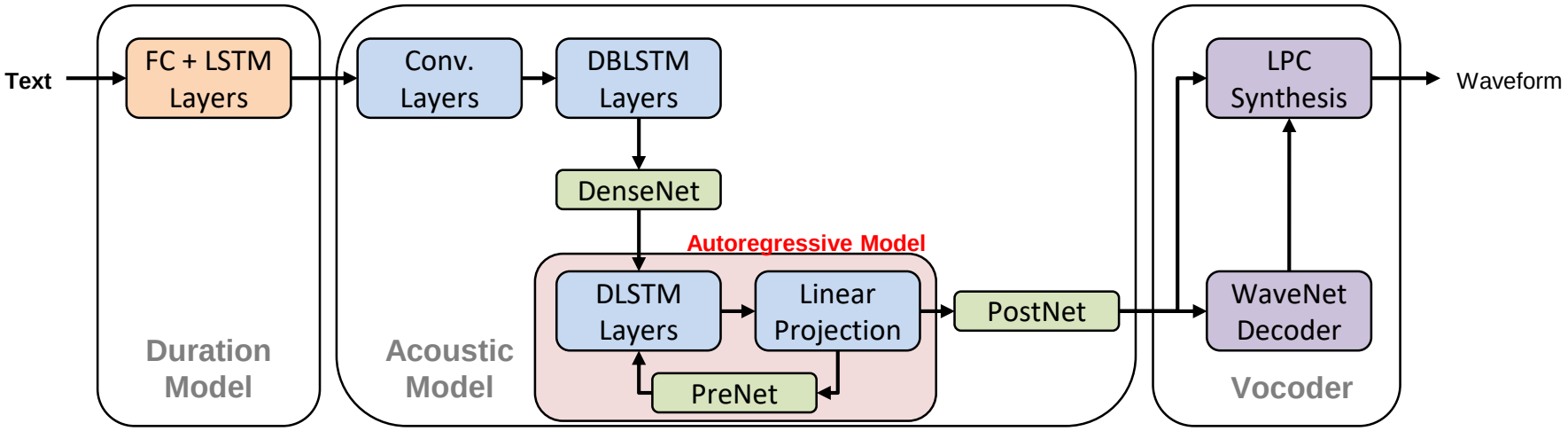
Acoustic Parameter 추정 정확도가 높아짐

CNN 모델이 음성 생성해줌

Attention 모델이 제대로 동작하지 않을 때, 합성음 품질에 Critical 영향을 줌

Tacotron 2

Clova HDTS



[HDTS 음성 합성 모델]

External duration model — 음소의 시간축 길이는 추정하고

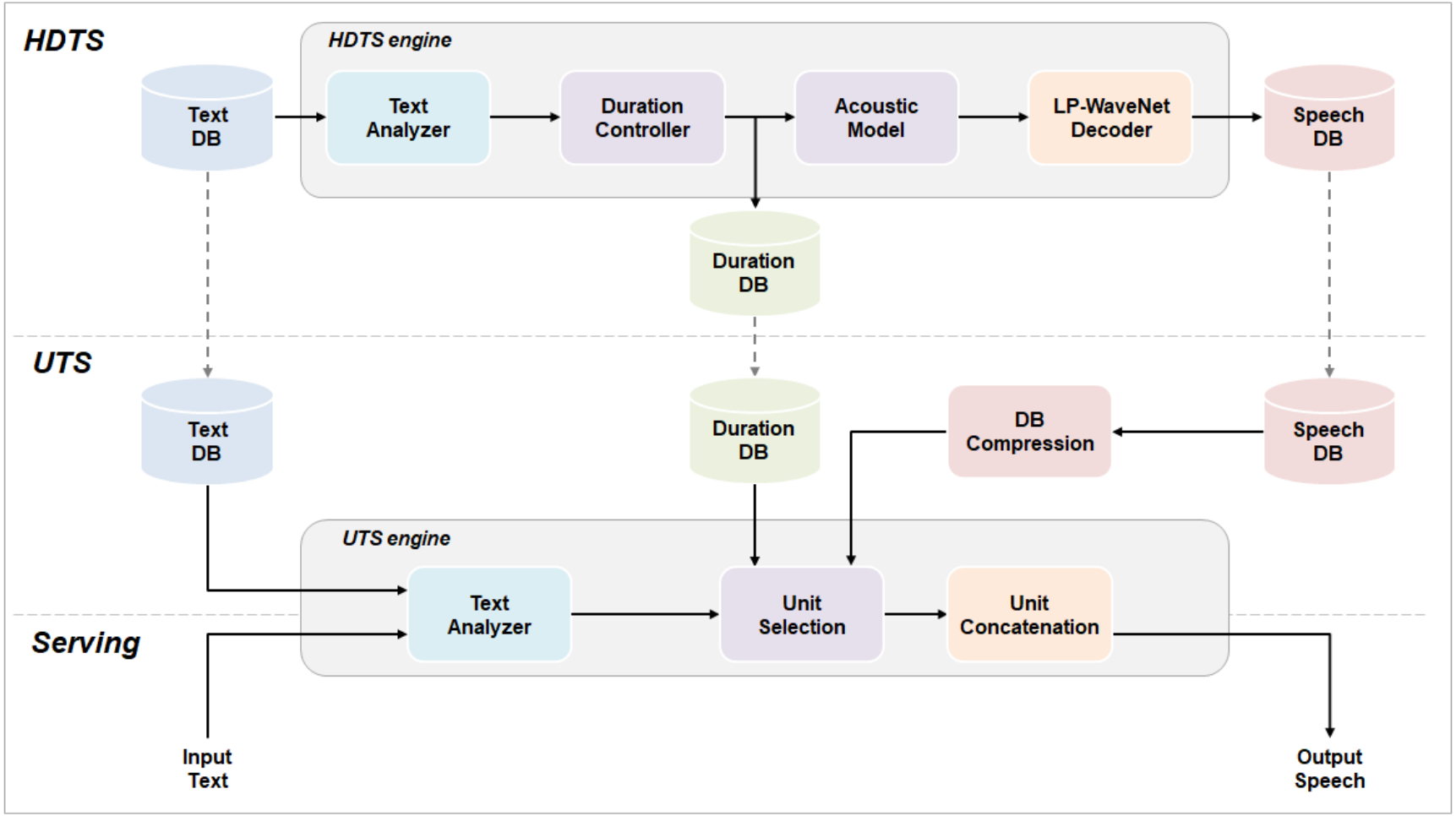
Autoregressive acoustic model — Tacotron decoder 구조는 유지하면서

LP-WaveNet vocoder — WaveNet 모델의 성능까지 올린다.

고품질의 음성을 안정적으로 생성할 수 있게됨

HDTS **Tacotron 2**

Clova HDTS



Text-to-speech

CLOVA Dubbing x NES

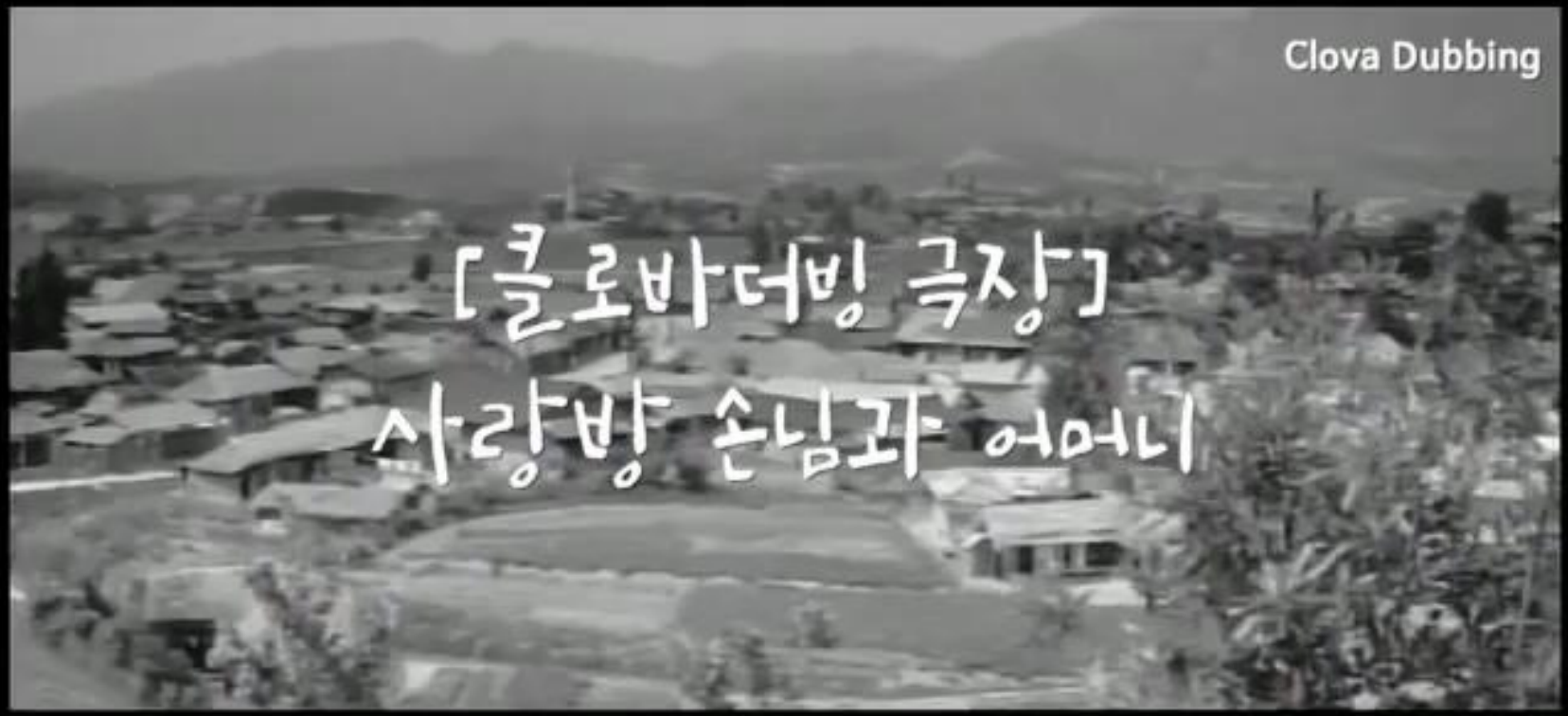
동영상에 보이스를 더하다

CLOVA Dubbing$^{\beta}$

자연스러운 클로바보이스로 동영상에
특별한 생동감을 더해주세요

클로바더빙 시작하기 >

글


Clova Dubbing
[클로바더빙 극장]
사랑방 손님과 어머니

Speaker-adaptive neural vocoders for parametric speech synthesis systems

Eunwoo Song[1], Jin-Seob Kim[1], Kyungguen Byun[2], Hong-Goo Kang[2]
[1]NAVER Corp., Seongnam, Korea
[2]Yonsei University, Seoul, Korea

NAVER CLOVA
YONSEI UNIVERSITY
DSP Lab.

Q / A