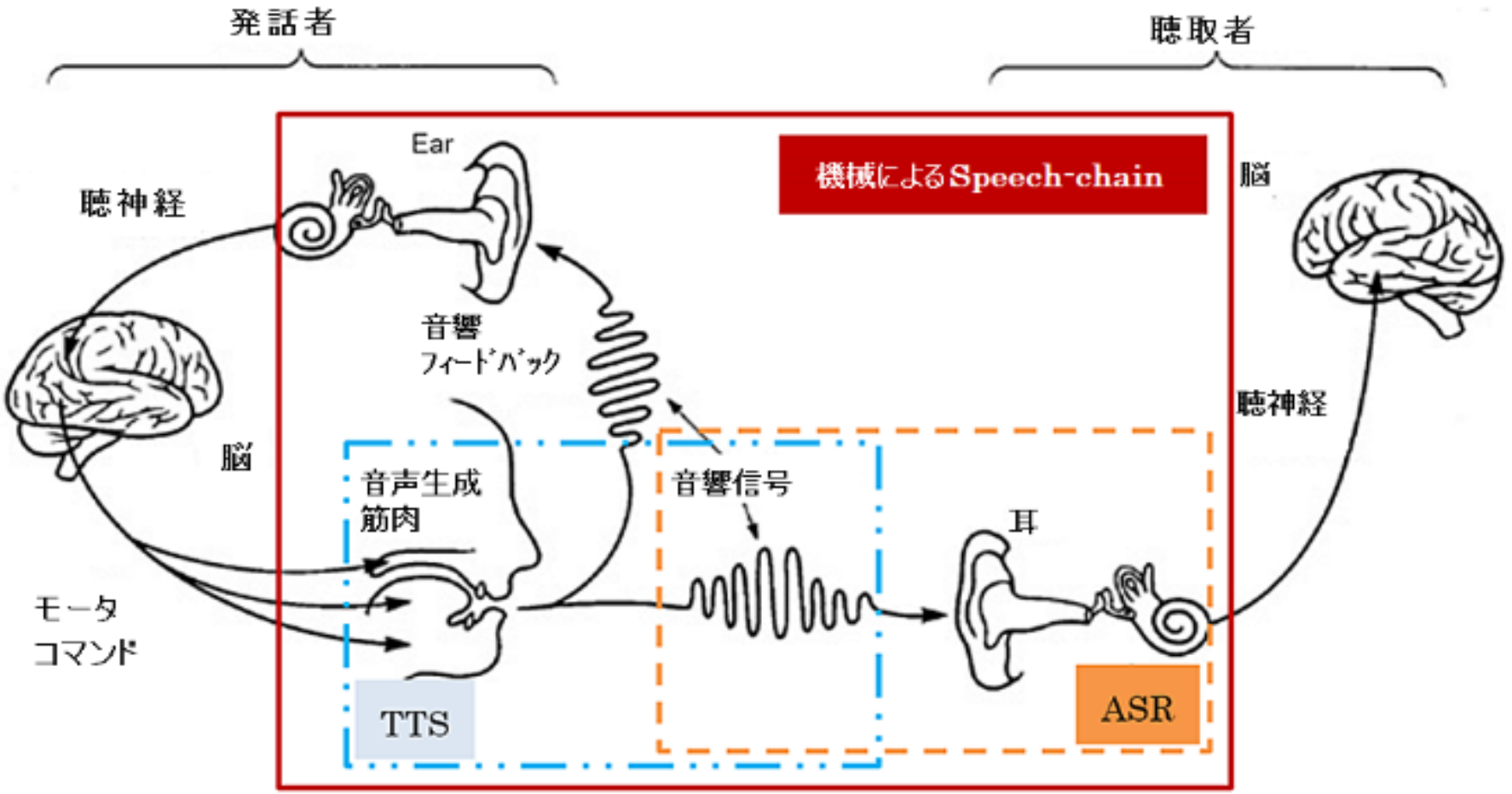
# Machine Speech Chainによる音声聴取生成システムのモデル化の試み

奈良先端科学技術大学院大学
中村　哲
with
Sashi Novitasari[1] and Sakriani Sakti[2,1]
[1]Nara Institute of Science and Technology, Japan
[2]Japan Advanced Institute of Science and Technology

# 人間の音声聴取生成におけるSpeech Chain

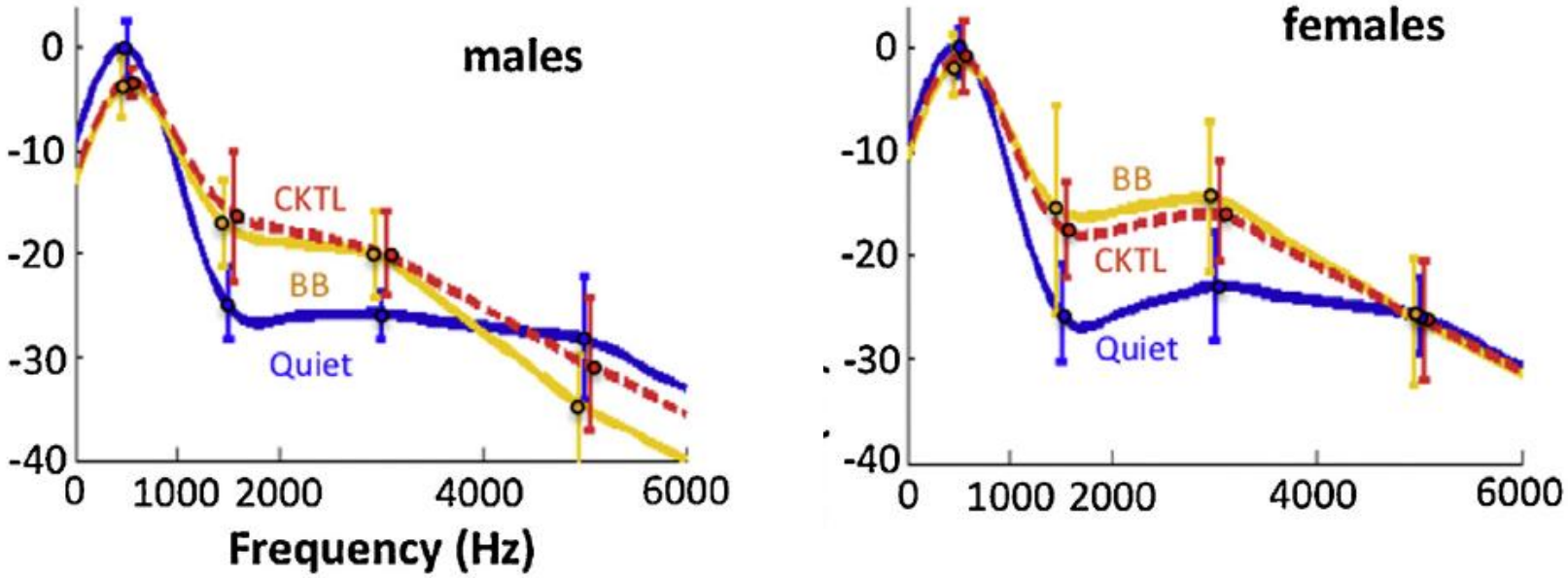
- 人間の音声生成と聴取の仕組み
  → 聴覚フィードバックを持つフィードバック系




引用＋追記： P. Denes and E. Pinson, The Speech Chain, ser. Anchor books. Worth Publishers, 1993.

# Lombard Effect

- Lombard Effect (Lombard, 1911)
  - Lombard Effect とは（Mokbel1992, Junqua1998, Garnier2014）
    雑音下で，
    - 聞こえるように音声のIntensityをあげる
    - 聞こえるようにf0，フォルマントを調整する（高い周波数方向にシフトする）
    - 時間軸の変調（発話速度，テンポ）を調整する




Garnier 2014から引用

# Delayed Auditory Feedback*1,2

- 遅延聴覚フィードバック(DAF)の吃音治療への適用:
  - DAF デバイス:自分の発話音声を僅かな時間遅延させてヘッドフォーンで再生
  - 吃音が治ったり, 吃音をバイパスすることができる

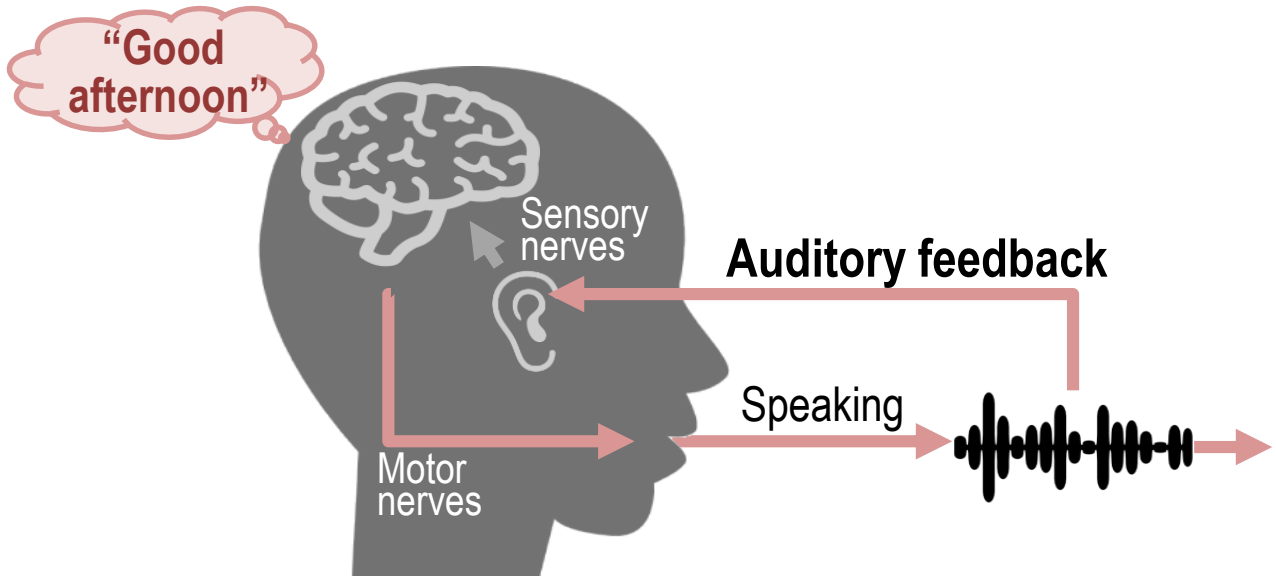
- 健常者に対するDAFの適用:
  - DAFは聴覚・発話生成システムの脳内メカニズムを調べるために利用される
  - 音声的現象:聴取されるDAF音の外乱に対するため発話者は発話速度を低下させたり, 音声の強調(振幅, F0)を行う
  - 言語的現象:音節の繰り返し, 発音間違い, 単語や単語の語尾の省略が生じる

*1Bernard S. Lee, "Delayed Speech Feedback", The Journal of the Acoustical Society of America **22**, 824 (1950);

# Machine Speech Chain

- **ねらい**
  → 人の処理のようなClosed-loop speech chain model を作る




**Machine Speech Chain**
**ASRとTTSを統合**

# 本日の内容

- Machine Speech Chain
  - 音声生成と聴取のフィードバックループ
  - 音声合成と音声認識の半教師あり(教師付き＋教師なし)学習
  - 多数話者 Machine Speech Chain
- 音声合成におけるMachine Speech Chain Inference
  - Machine Speech ChainによるLombard音声合成
- 漸進的Incremental Speech Chain Inference
  - 動的なMachine Speech Chain Inference
- まとめ

# 本日の内容

- Machine Speech Chain
  - 音声生成と聴取のフィードバックループ
  - 音声合成と音声認識の半教師あり(教師付き＋教師なし)学習
  - 多数話者 Machine Speech Chain
- 音声合成におけるMachine Speech Chain Inference
  - Machine Speech ChainによるLombard音声合成
- 漸進的Incremental Speech Chain Inference
  - 動的なMachine Speech Chain Inference
- まとめ

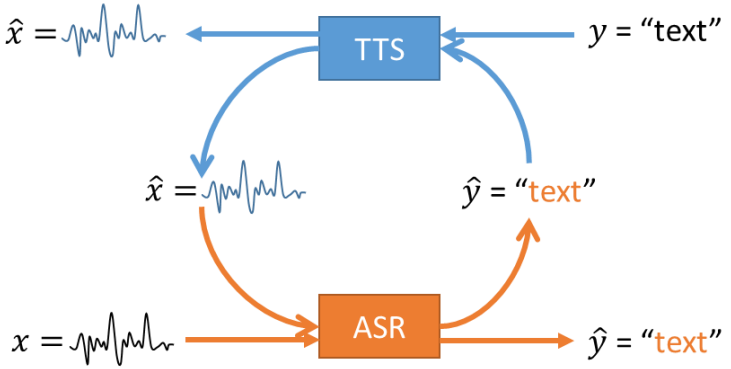
# これまでの音声認識・合成

- 音声認識：
  - モデル学習：文字誤り率，単語誤り率を低下させるように，最尤推定，識別学習
    - 音声と正解スクリプトのペアデータによる教師付き学習
    - テスト環境の音声を収集してモデル適応
  - 音声認識時：
    - 学習済み（あるいは適応済み）モデルを用いて音声認識
    - 雑音などがあると信号処理をしてモデル音声に近づけて使用
- 音声合成：
  - モデル学習：正解音声と生成音声の信号，スペクトルの誤差を最小化
    - テキストと自然音声のペアデータによる教師付き学習
  - 音声合成時：
    - 学習済み（あるいは適応済み）モデルを用いて音声合成

- これまでは音声認識と合成は別の研究？

# Machine Speech Chain

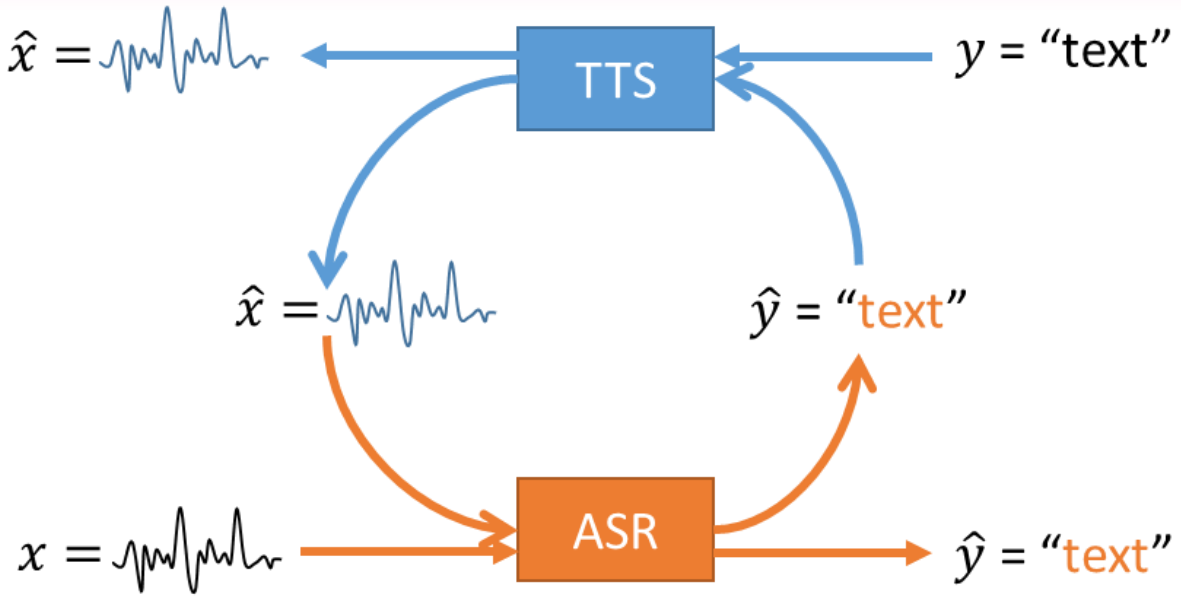
- Machine Speech Chain:
  - End-to-end 深層学習ベースの音声認識，音声合成モデルの登場
    - 殆ど同様のモデル　注意機構付きEncoder-decoderモデル，Transformerモデル
    - 誤差逆伝搬法により学習できる
  - Machine Speech Chain：
    - End-to-end 音声認識，音声合成モデルを連結し誤差を伝搬して学習
    - 半教師付き学習，自己学習




**Andros Tjandra, Sakriani Sakti, Satoshi Nakamura, "Listening while Speaking: Speech Chain by Deep Learning", IEEE ASRU 2017**

# Machine Speech Chain




- 定義:
  - $x$ = 学習用音声データ, $y$ = 学習用テキストデータ
  - $\hat{x}$ = 再構築された予測音声データ, $\hat{y}$ = 再構築された予測テキストデータ
  - $ASR(x): x \rightarrow \hat{y}$ (seq2seq によるASRで音声をテキスト化)
  - $TTS(y): y \rightarrow \hat{x}$ (seq2seq によるTTSでテキストを音声化)

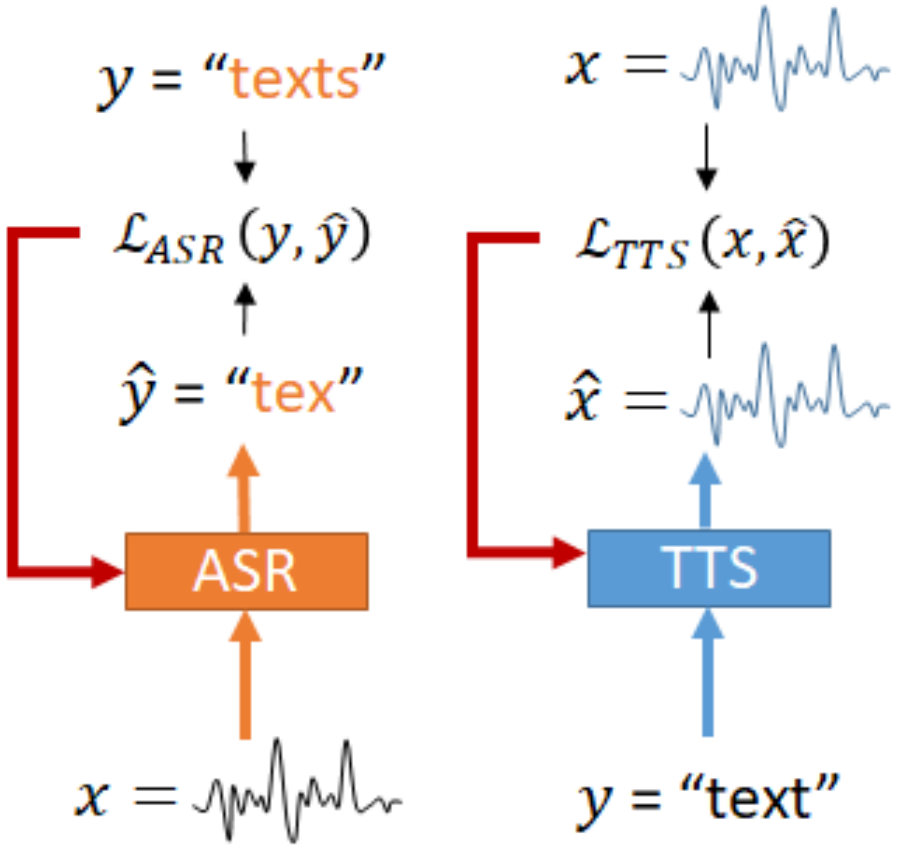
# Machine Speech Chain

## Case #1: 音声とテキストによる教師付き学習

- **音声とテキストのペアデータ$(x, y)$が存在**
  - ASR, TTSモデルを教師付き学習
  - 直接それぞれのモデルを学習:

→ $ASR$ by minimize $\mathcal{L}_{ASR}(y, \hat{y})$
→ $TTS$ by minimizing loss between $\mathcal{L}_{TTS}(x, \hat{x})$

**ASRとTTSは別々に学習される**

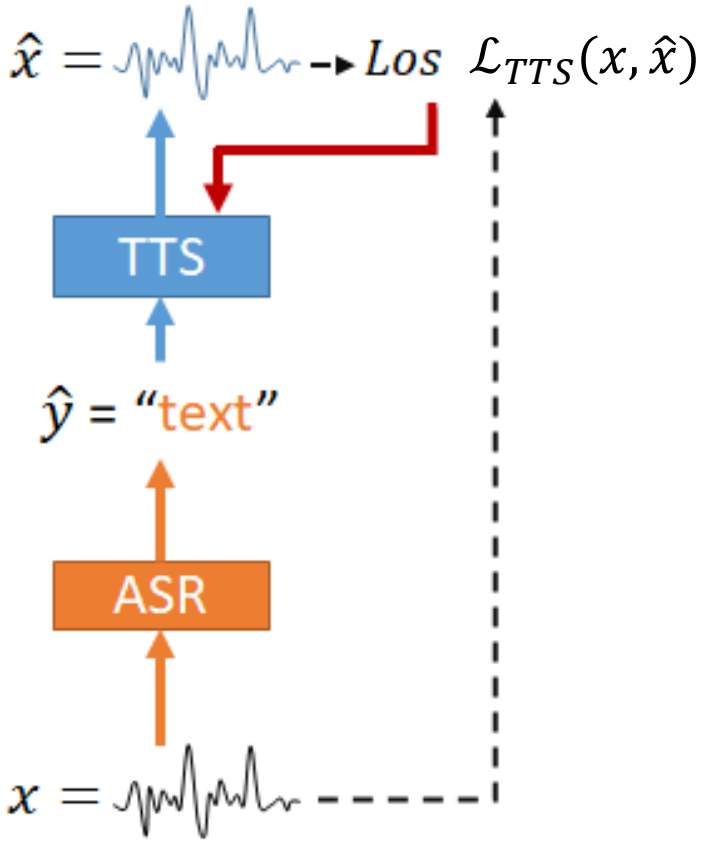


# Machine Speech Chain

## Case #2: 音声データのみを用いた半教師付き学習

– 書き起こしのない音声$x$だけがある場合

1. ASRにより音声認識仮説 $\hat{y}$ を生成
2. $\hat{y}$を用いてTTSを行い音声特徴を予測（再構成）$\hat{x}$
3. 原音声$x$ と予測音声$\hat{x}$の損失誤差 $\mathcal{L}_{TTS}(x, \hat{x})$ を求める




**TTSモデルがASRを利用して更新される**

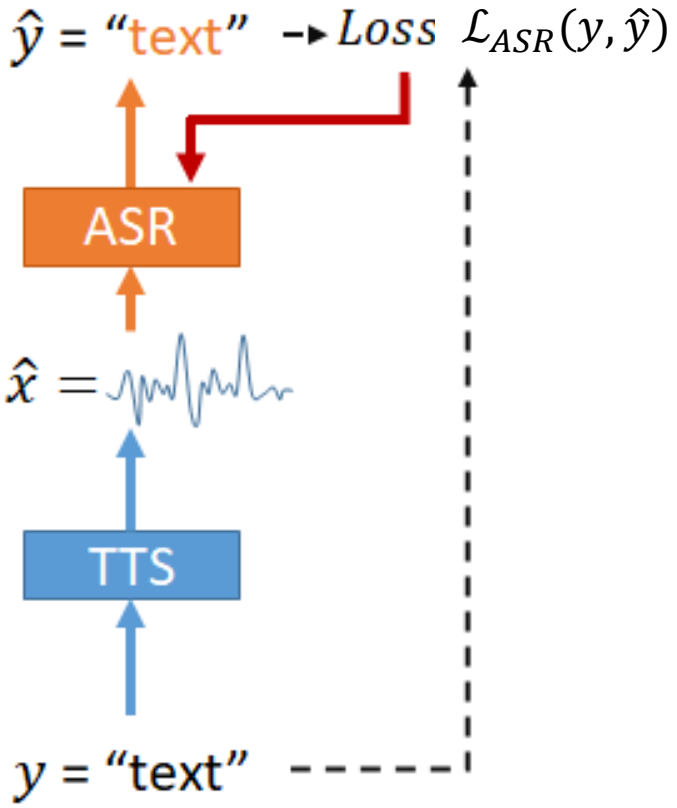
# Machine Speech Chain

## Case #3: テキストのみを用いた半教師付き学習

– 音声はなくテキスト $y$ のみがある場合

1. TTS により音声特徴を生成 $\hat{x}$
2. $\hat{x}$をASRし予測(再構成)テキスト $\hat{y}$ を再構成
3. 原テキスト $y$ と予測テキスト $\hat{y}$ の文字誤り損失 $\mathcal{L}_{ASR}(y, \hat{y})$ を求める

**ASRモデルがTTSを利用して更新される**



# Machine Speech Chainの損失関数

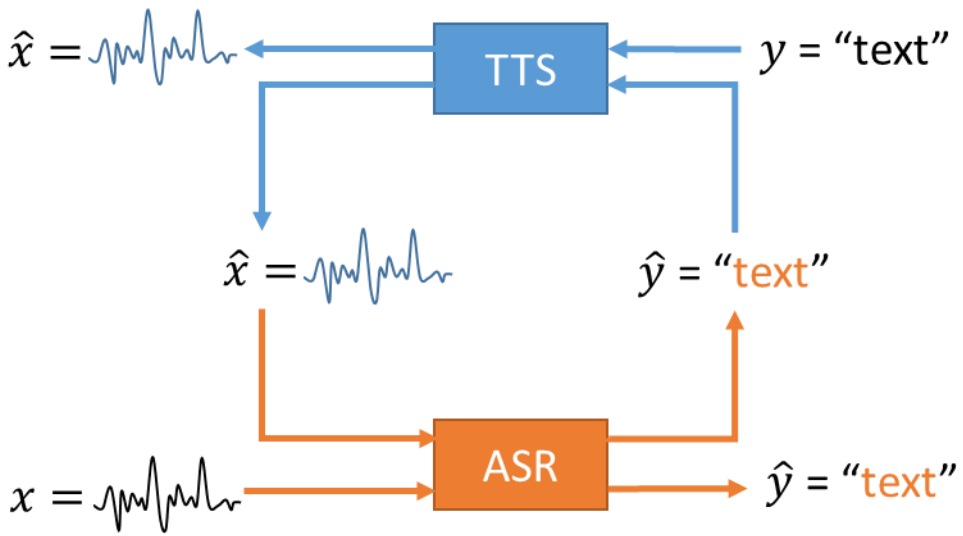
- Combined loss:

$$\ell_{ALL} = \underbrace{\alpha\left(\ell_{TTS}^{P} + \ell_{ASR}^{P}\right)}_{\text{教師付きデータ損失}} + \underbrace{\beta\left(\ell_{TTS}^{U} + \ell_{ASR}^{U}\right)}_{\text{教師なしデータ損失}}$$

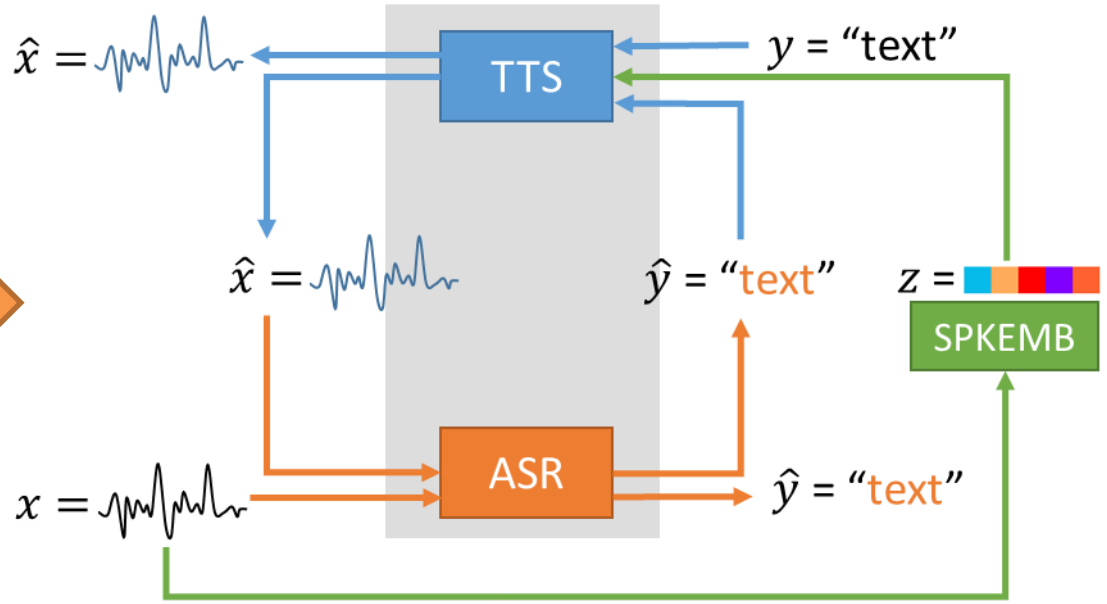
$\alpha,\beta$ はハイパーパラメータ

# 多数話者 Machine Speech Chain

一人の声の音声合成と音声認識のモデル




Speaker Embedding（SPKEMB）を導入する.
TTS時にもSPKEMBからベクトルを生成し多様な音声を生成.




**Andros Tjandra, Sakriani Sakti, Satoshi Nakamura, “Machine Speech Chain with One-shot Speaker Adaptation”, INTERSPEECH 2018**

# 実験結果 WSJ

Table 1: *Character error rate (CER (%)) comparison between results of supervised learning and those of a semi-supervised learning method, evaluated on test_eval92 set*

| Model | CER (%) |
|---|---|
| **Supervised training: WSJ *train_si84* (paired) → Baseline** | |
| Att Enc-Dec [19] | 17.01 |
| Att Enc-Dec [20] | 17.68 |
| Att Enc-Dec (ours) | 17.35 |
| **Supervised training: WSJ *train_si284* (paired) → Upperbound** | |
| Att Enc-Dec [19] | 8.17 |
| Att Enc-Dec [20] | 7.69 |
| Att Enc-Dec (ours) | 7.12 |
| **Semi-supervised training: WSJ *train_si84* (paired) + *train_si200* (unpaired)** | |
| Label propagation (greedy) | 17.52 |
| Label propagation (beam=5) | 14.58 |
| **Proposed speech chain (Sec. 2)** | **9.86** |

Table 2: *L2-norm squared on log-Mel spectrogram to compare the supervised learning and those of a semi-supervised learning method, evaluated on test_eval92 set. Note: We did not include standard Tacotron (without SPKREC) into the table since it could not output various target speaker.*

| Model | L2-norm$^2$ |
|---|---|
| **Supervised training: WSJ *train_si84* (paired) → Baseline** | |
| Proposed Tacotron (Sec. 4) (ours) | 1.036 |
| **Supervised training: WSJ *train_si284* (paired) → Upperbound** | |
| Proposed Tacotron (Sec. 4) (ours) | 0.836 |
| **Semi-supervised training: WSJ *train_si84* (paired) + *train_si200* (unpaired)** | |
| **Proposed speech chain (Sec. 2 + Sec. 4)** | **0.886** |

# 本日の内容

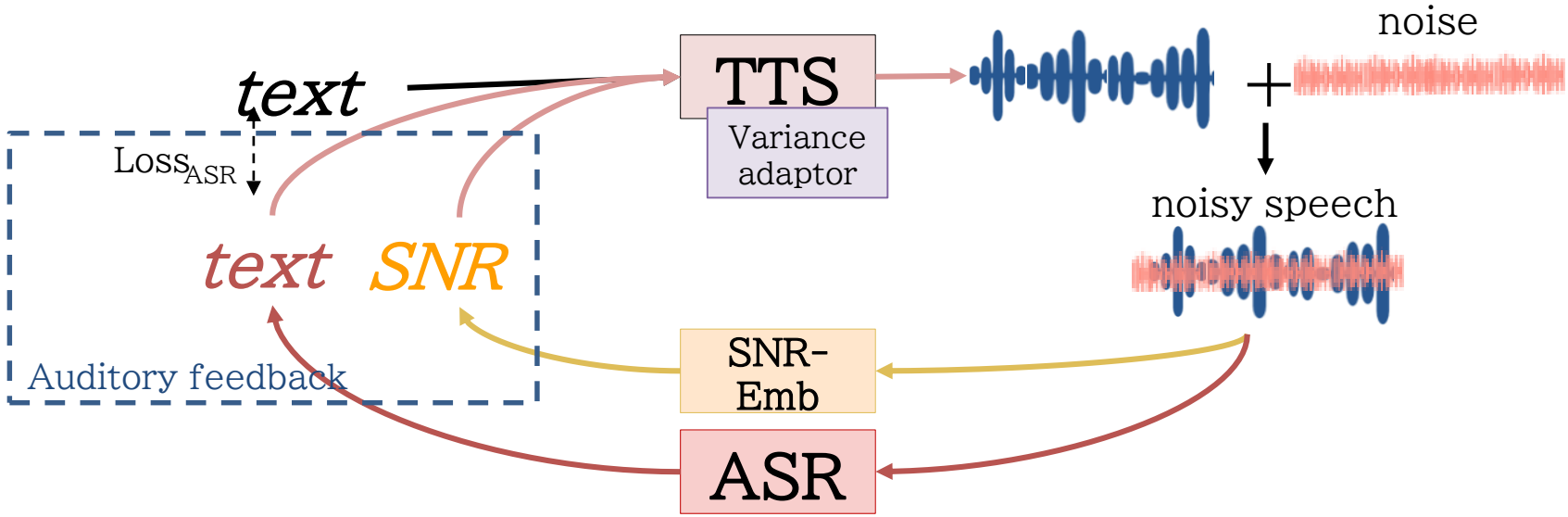
- Machine Speech Chain
  - 音声生成と聴取のフィードバックループ
  - 音声合成と音声認識の半教師あり(教師付き＋教師なし)学習
  - 多数話者 Machine Speech Chain
- 音声合成におけるMachine Speech Chain Inference
  - Machine Speech ChainによるLombard音声合成
- 漸進的Incremental Speech Chain Inference
  - 動的なMachine Speech Chain Inference
- まとめ

# Speech Chainによるフィードバック

TTSの際に合成音を聴取してTTSの品質を動的に調整する → Lombard Speech

ASRとSNR測定を行ってTTSの発話を変化させる

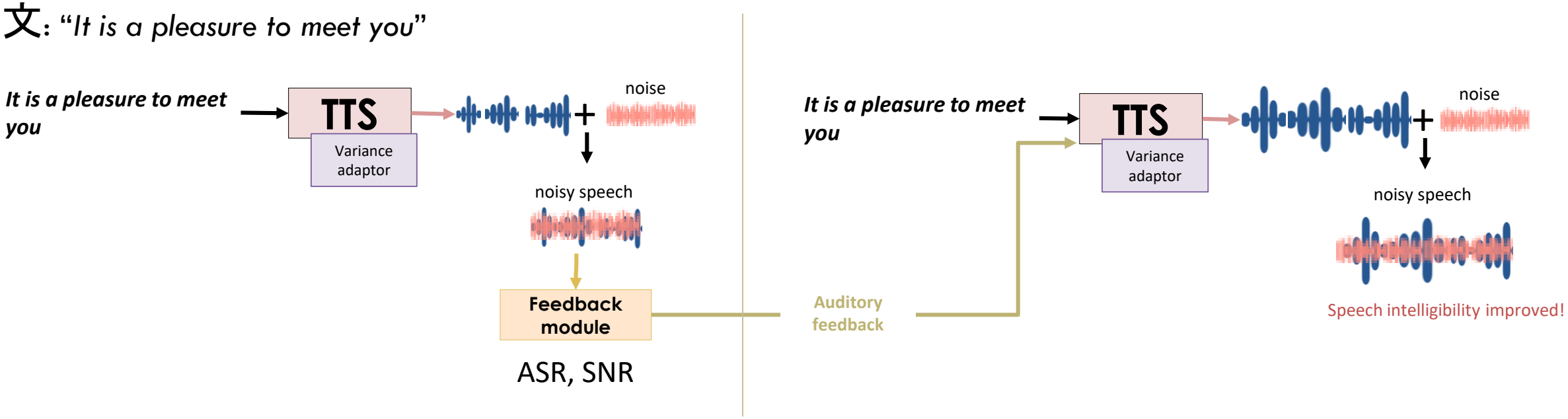



Sashi Novitasari, Sakriani Sakti, and Satoshi Nakamura, "Dynamically Adaptive Machine Speech Chain Inference for TTS in Noisy Environment: Listen and Speak Louder", INTERSPEECH2021

# システムの構成

- 通常のTTS　　：文から音声を合成
- 提案法のTTS　：
  - Step1: 文から音声を合成する
  - Step2: SNR測定と音声認識を行いそのフィードバックを用いて音声を再合成

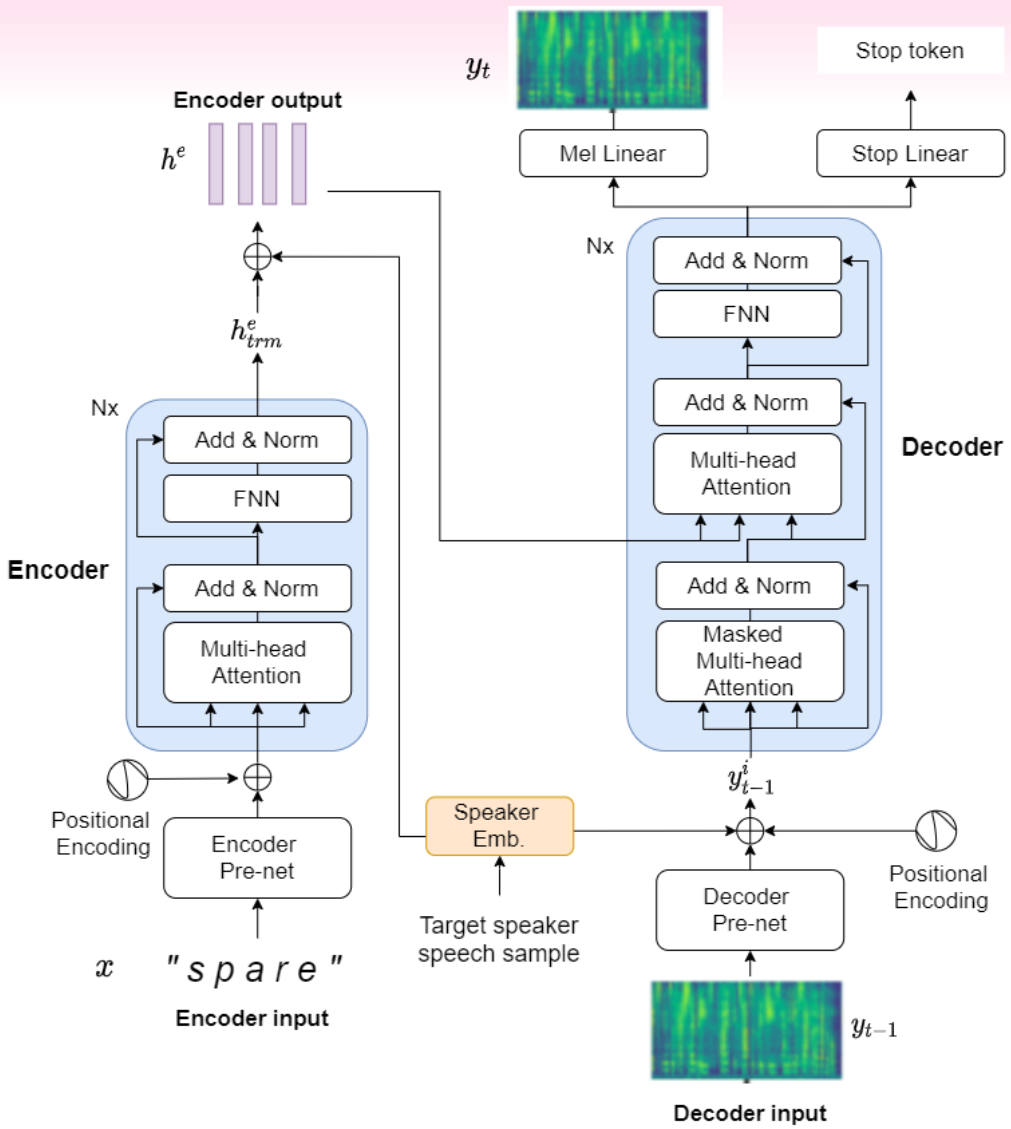


# フィードバックを有するTransformerベースのTTS

- 基本TTS構成： Transformer TTS [Li et el., 2018]
  - 入力 ： 文字列
  - 出力 ： 音声のスペクトル特徴
    (80 dims. Mel-spectrogram)

    - Multi-speaker 条件
      - Multi-speaker TTS Transformer [Chen et al., 2020]
      - Speaker embedding: Deep Speaker [Li et al., 2017] (similar to TTS in the basic machine speech chain)

- **提案法TTSの構造(3種類を比較してみた)**
  - a) TTS + SNR embedding のみ
  - b) TTS + ASRとSNR embedding
  - c) TTS + ASRとSNR embedding,
    さらに Variance adaptor




Multi-speaker Transformer TTS with Deep Speaker embedding

# SNR embeddingを利用したフィードバック

SNRによるフィードバック

- SNR embedding($Z_{SNR}$): 雑音環境中で合成された音声 ($y^{noisy}$)のSNR

$$Z_{SNR} = SNR\ Emb\ (y^{noisy})$$

  - SNR識別モデルは予め学習しておく

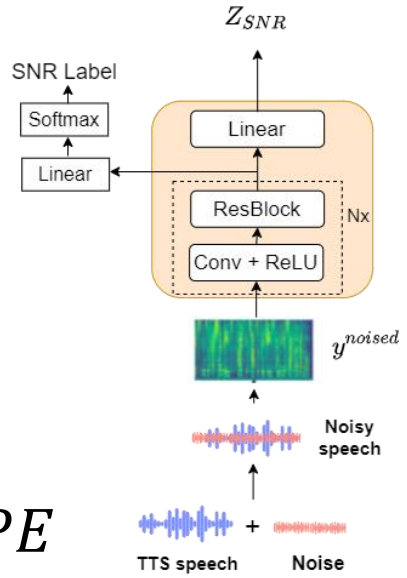
- フィードバックの手順
  - Encoder の出力に加える ($h^e$)

$$h^e = h^e_{trm} + Z_{SPK} + Z_{SNR}$$

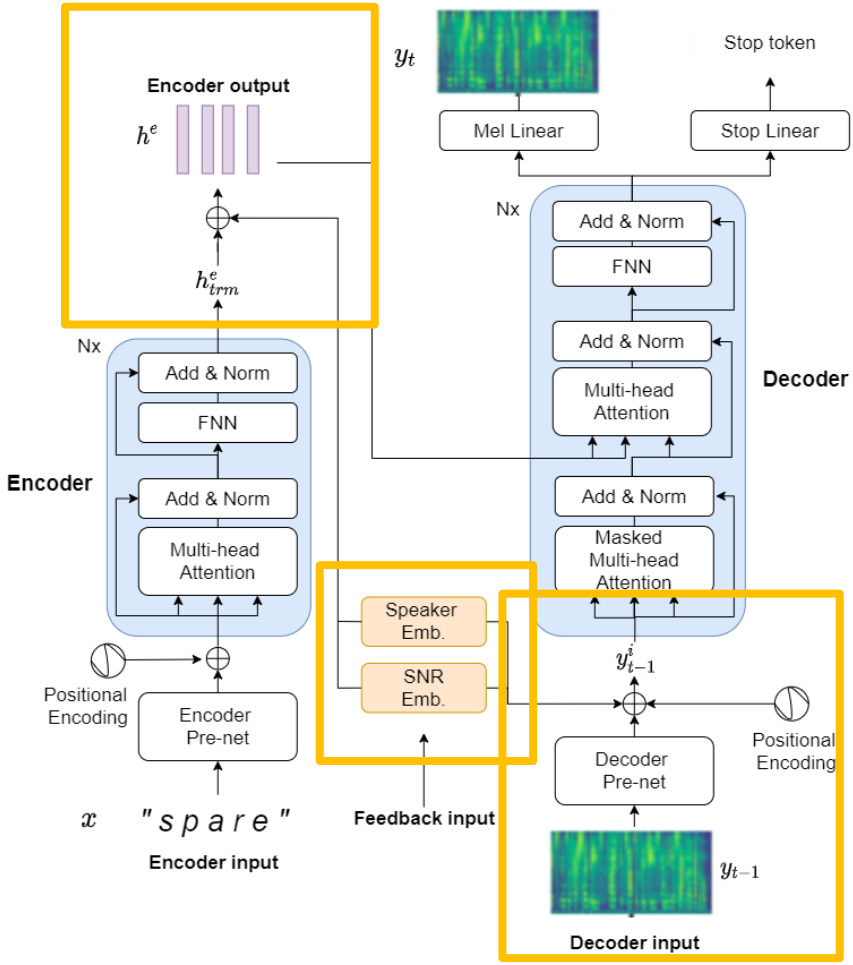
  - Decoderの第一層への入力 ($y^i_{t-1}$)

$$y^i_{t-1} = prenet(y_{t-1}) + Z_{SPK} + Z_{SNR} + PE$$

$Z_{SPK}$ : speaker embedding
$PE$ : positional encoding




SNR emb. module




Transformer TTS with SNR emb.

# SNR とASR-loss embeddingによるフィードバック

フィードバック:

- SNR embedding:
- ASR-loss embedding ($Z_{ASR}$):
  ASRの誤りをEmbedding する

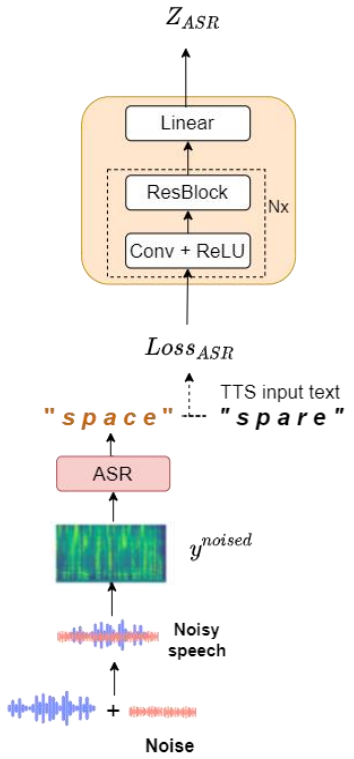
$$Z_{ASR} = ASR\ Loss\ Emb\left(Loss_{ASR}(x, p_x)\right)$$
$$p_x = p(x|y^{noisy})$$

$x$ : TTS input text (correct text)
$p_x$ : ASR hypothesis

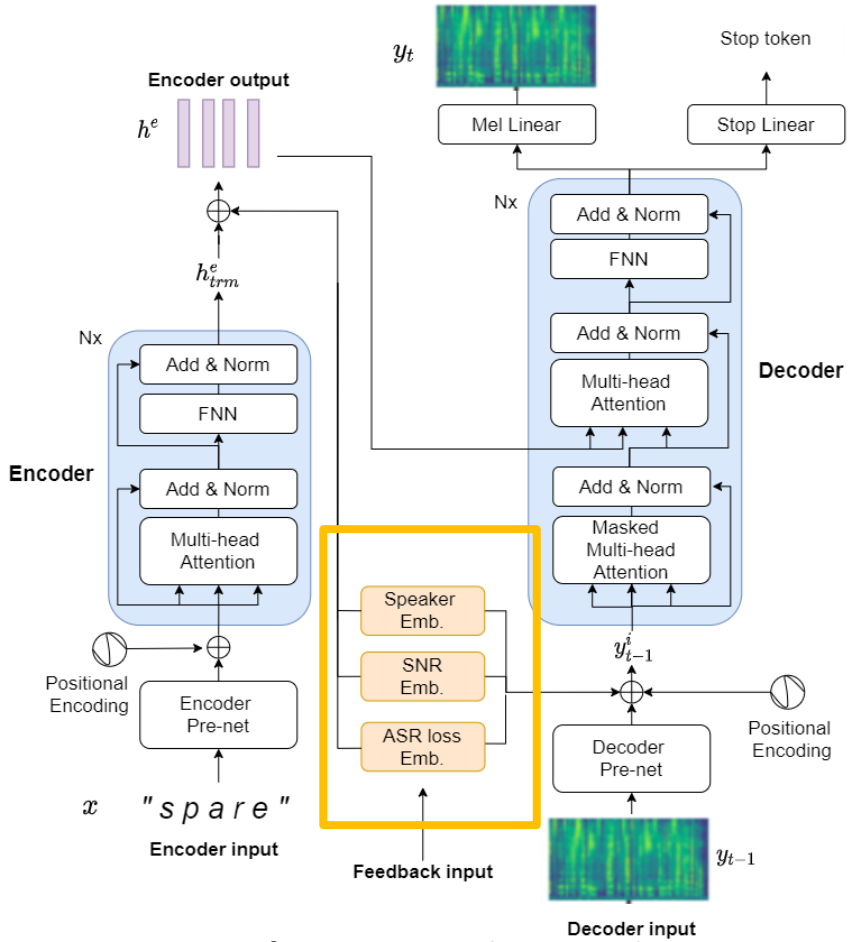
- Encoderの出力とDecoderの入力に加える

$$h^e = h^e_{trm} + Z_{SPK} + Z_{SNR} + Z_{ASR}$$

$$y^i_{t-1} = prenet(y_{t-1}) + Z_{SPK} + Z_{SNR} + Z_{ASR} + PE$$




ASR-loss emb. module




Transformer TTS with SNR and ASR-loss emb.

# TTS with SNR, ASR-loss embedding, Variance adaptor

Variance Adaptor を用いて韻律を制御

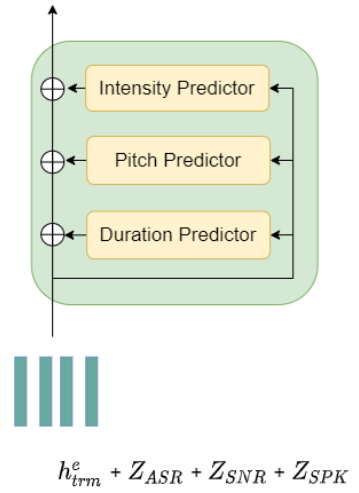
- Fast Speech [Ren et al., 2020]で提案された方法，ここではAR型のTransformerに修正して適用
- Intensity, Pitch, Durationの3つを制御：
  - 文字レベルの韻律を制御できる

$$v^X = Predictor^X(h^e_{trm} + Z_{SPK} + Z_{SNR} + Z_{ASR})$$

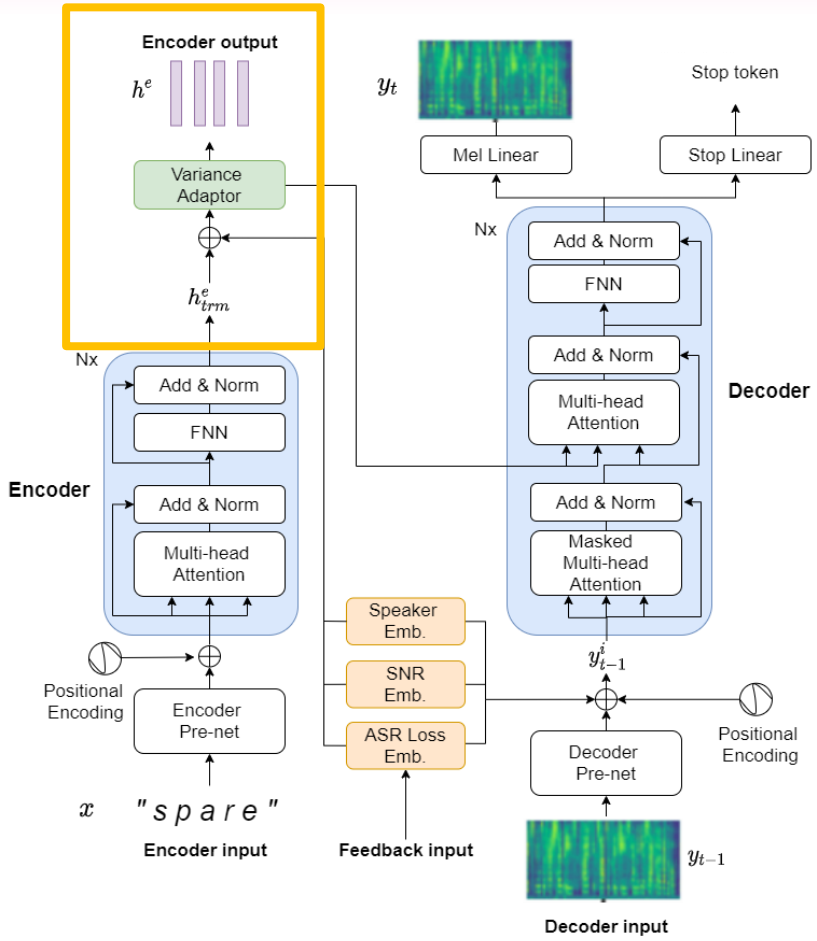
  - Intensity predictor $(X = G)$
  - Pitch predictor $(X = P)$
  - Duration predictor $(X = D)$

- Encoder出力に韻律情報を加える

$$h^e = v^G + v^P + v^D + (h^e_{trm} + Z_{SPK} + Z_{SNR} + Z_{ASR})$$




Variance adaptor
Intensity, Pitch, Duration



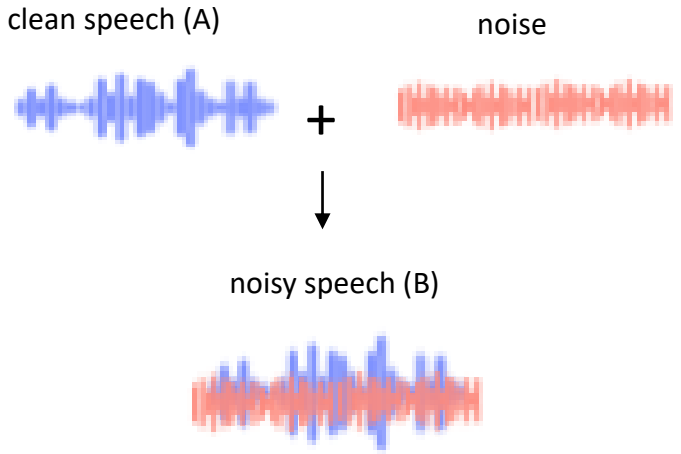

Transformer TTS with SNR, ASR-loss embedding, and variance adaptor

# TTS実験に利用したデータ

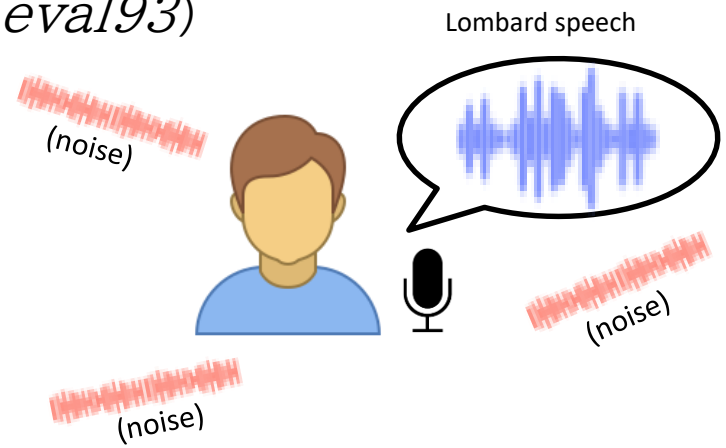
A. Wall Street Journal (WSJ) speech [Paul et al., 1992]
- 多数話者音声　81時間
- 学習セット: *SI-284* set, 開発セット: *dev92* set,
  評価セット: *eval93* set

B. 雑音付加WSJ speech
- オリジナルのWSJ speech に雑音を付加
  - Noise type：白色雑音, バブル雑音
  - SNR条件　　：SNR 0 と SNR -10




C. 自然な Lombard 音声
- 話者1名によるクリーン, および, 雑音環境下の音声
- 読み上げテキスト: WSJ speech transcription（*dev92 + eval93*）

D. WSJ speechを提案法で合成した疑似Lombard音声
- Intensity, Pitch, Duration をフィードバックにより
  修正した合成音
  (Natural Lombard − Natural) + WSJ



# 明瞭度(CER) 実験結果

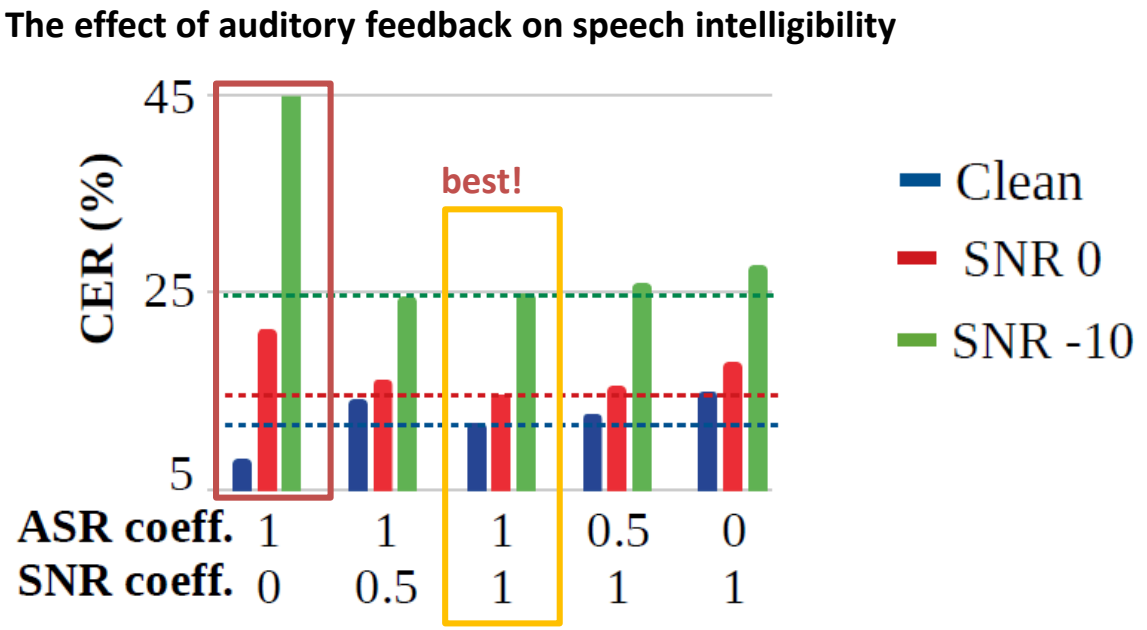
- 評価：音声明瞭度をASR 文字誤り率（CER）で評価. TTS音声を認識
- 提案法TTSフィードバック：４回
- 提案法：TTS + SNR-ASR loss emb. + variance adaptor ベスト
  - SNR, ASRフィードバックが有
  - Variance adaptor が文字レベルの韻律を制御
- さらに改良の余地がある

**Speech intelligibility measure (CER %) at different SNR levels using ASR trained on clean and noisy conditions.**

| System | Clean | SNR 0 | SNR -10 |
|---|---|---|---|
| **Baseline TTS** | | | |
| Standard TTS | 18.32 | 70.54 | 77.07 |
| + modification into Lombard speech | 18.32 | 44.68 | 57.86 |
| + Fine-tuning with Lombard speech | 13.40 | 28.12 | 46.13 |
| **Proposed TTS** | | | |
| TTS + SNR emb. | **11.58** | 22.82 | 42.00 |
| TTS + SNR-ASR loss emb. | 12.55 | 16.11 | 25.61 |
| TTS + SNR-ASR loss emb. + var. adaptor | 11.99 | **14.70** | **24.96** |
| **Topline (human natural speech)** | | | |
| Natural speech | 7.43 | 22.17 | 58.81 |
| + modification into Lombard speech | 7.43 | 13.24 | 15.15 |
| Natural Lombard speech | 7.43 | 11.46 | 20.56 |

# フィードバックの考察

- ASR embedding と SNR embeddingに重みを入れて効果を調査




- クリーン条件では，ASRフィードバックのみがよい (ASR coeff 1, SNR coeff 0)
- 雑音条件では，ASR＋SNRフィードバックがよい(ASR coeff 1, SNR coeff 1)

SNR, ASR フィードバックの両方が必要

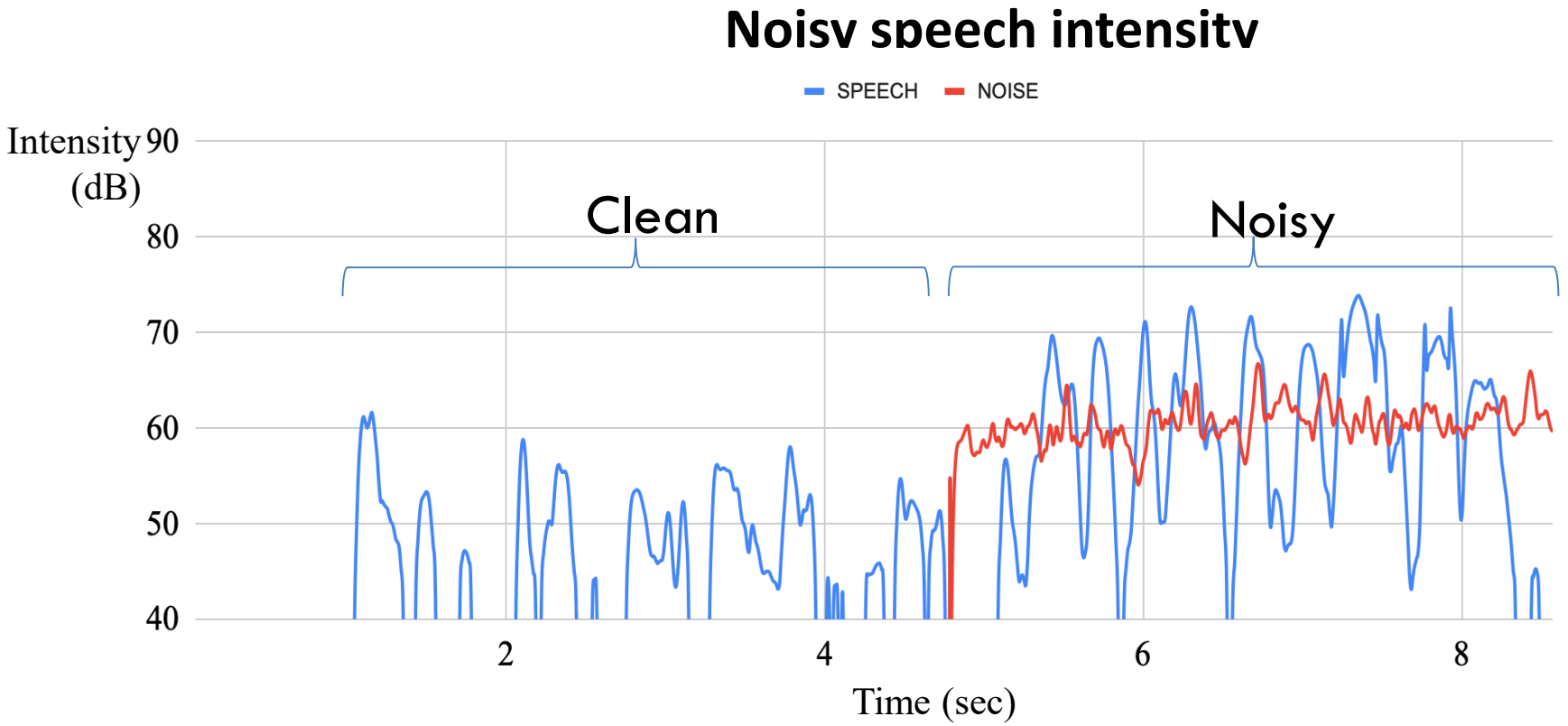
# 本日の内容

- Machine Speech Chain
  - 音声生成と聴取のフィードバックループ
  - 音声合成と音声認識の半教師あり(教師付き＋教師なし)学習
  - 多数話者 Machine Speech Chain
- 音声合成におけるMachine Speech Chain Inference
  - Machine Speech ChainによるLombard音声合成
- **漸進的Incremental Speech Chain Inference**
  - **動的なMachine Speech Chain Inference**
- まとめ

# 雑音中の発話

実際のヒトの発話は雑音が大きくなると大きくなる
- 人の反応速度 : 90-176 ms [Foery, 2008]

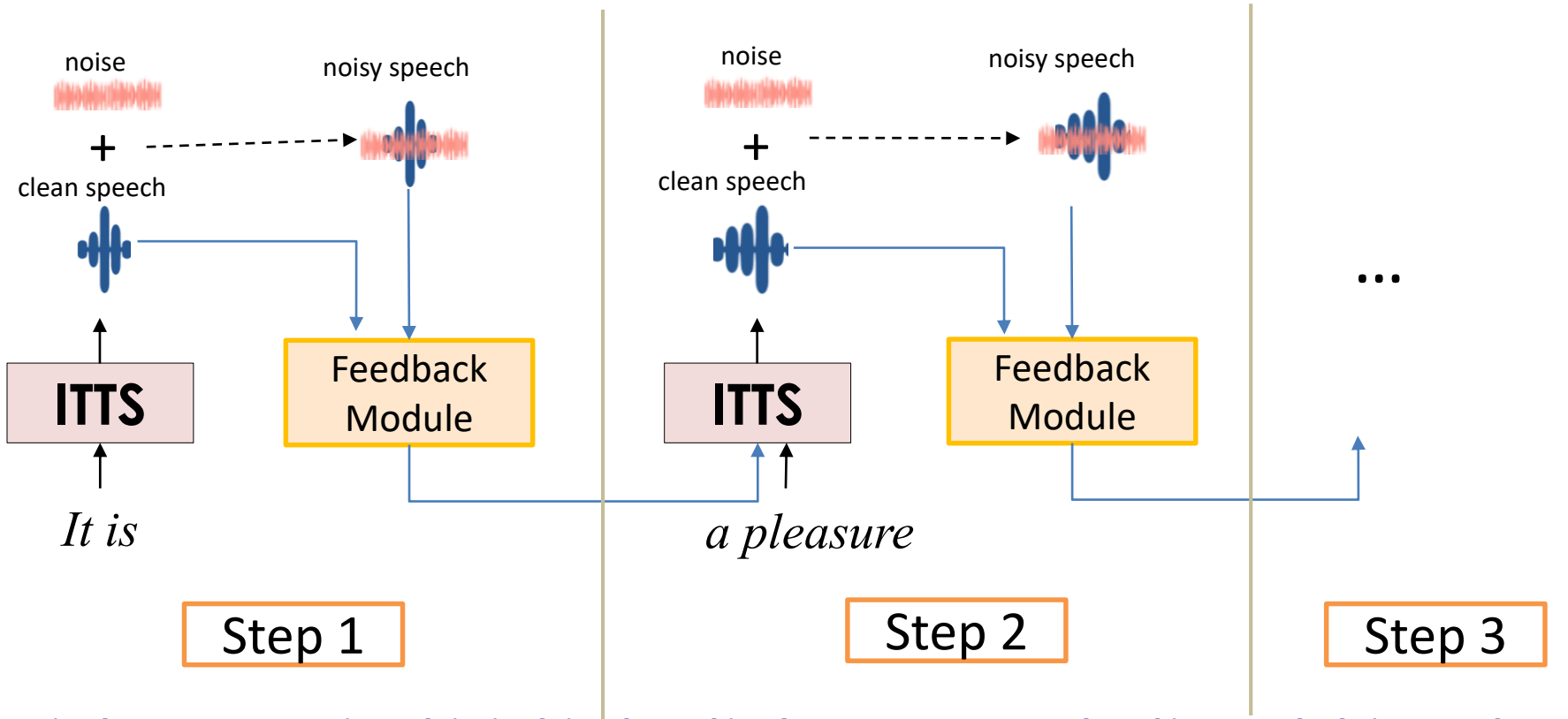



漸進的，動的にフィードバックしてTTSを変化させる必要がある

# Machine Speech Chain Inferenceを有する漸進的TTS

## 一定の窓幅でステップワイズにフィードバック処理を行う

Full sentence: "*It is a pleasure to meet you*"　下記の例ではステップごとに2単語を処理




残念ながら
ここから先は
未発表…

Sashi Novitasari, Andros Tjandra, Tomoya Yanagita, Sakriani Sakti and Satoshi Nakamura, "Incremental Machine Speech Chain Towards Enabling Listening while Speaking in Real-time", INTERSPEECH 2020
Tomoya Yanagita, Sakriani Sakti and Satoshi Nakamura, "Neural iTTS: Toward Synthesizing Speech in Real-time with End-to-end Neural Text-to-Speech Framework", 10th Speech Synthesis Workshop (SSW10) , Sep. 2019

# まとめ

- 本日の内容
  - Machine Speech Chain
    - 音声生成と聴取のフィードバックループ
    - 音声合成と音声認識の半教師あり(教師付き＋教師なし)学習
  - 音声合成におけるMachine Speech Chain Inference
    - Machine Speech ChainによるLombard音声合成
  - 漸進的Incremental Speech Chain Inference
    - 動的なMachine Speech Chain Inference
- 今後の研究, および関連研究
  - リアルタイムフィードバックを持つMachine Speech Chain によるTTS,ASRの同時学習, 推論
  - マルチモーダルMachine Speech Chain
  - Zero-speech Challenge
  - 音声言語処理中の脳活動の計測と活用