END-TO-END FEEDBACK LOSS IN SPEECH CHAIN FRAMEWORK VIA STRAIGHT-THROUGH ESTIMATOR

Andros Tjandra[1], Sakriani Sakti[1,2], Satoshi Nakamura[1,2]

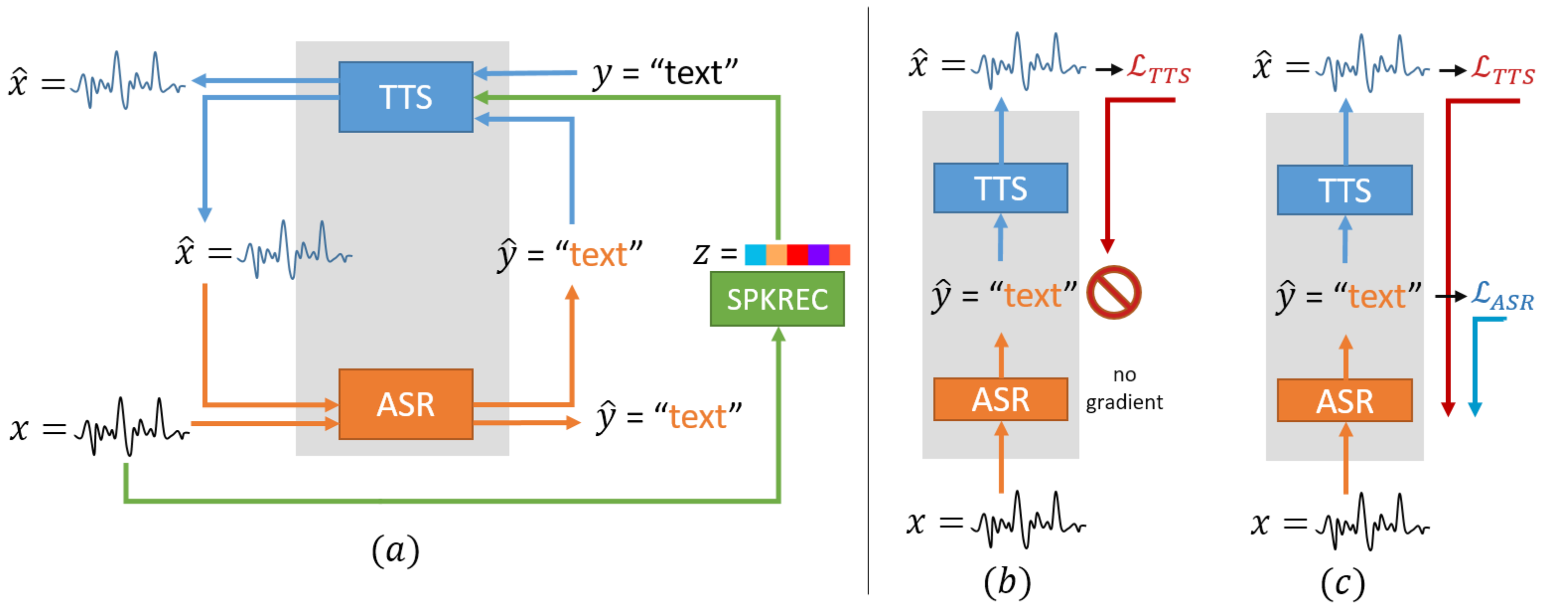
1) Nara Institute of Science and Technology, Japan 2) RIKEN, Center for Advanced Intelligence Project AIP, Japan

✉ {andros.tjandra.ai6, ssakti, s-nakamura}@is.naist.jp

1. Introduction

- Speech chain model integrates ASR and TTS into a single cycle during training.
- By combining both models, we could train with auxiliary feedback loss.
- Problem:
  - The output from ASR are discrete tokens
  - Non-differentiable (ASR → TTS)
- Solution:
  - Apply straight-through estimator on Gumbel-softmax or $argmax$ sample

2. Speech Chain and Feedback loss



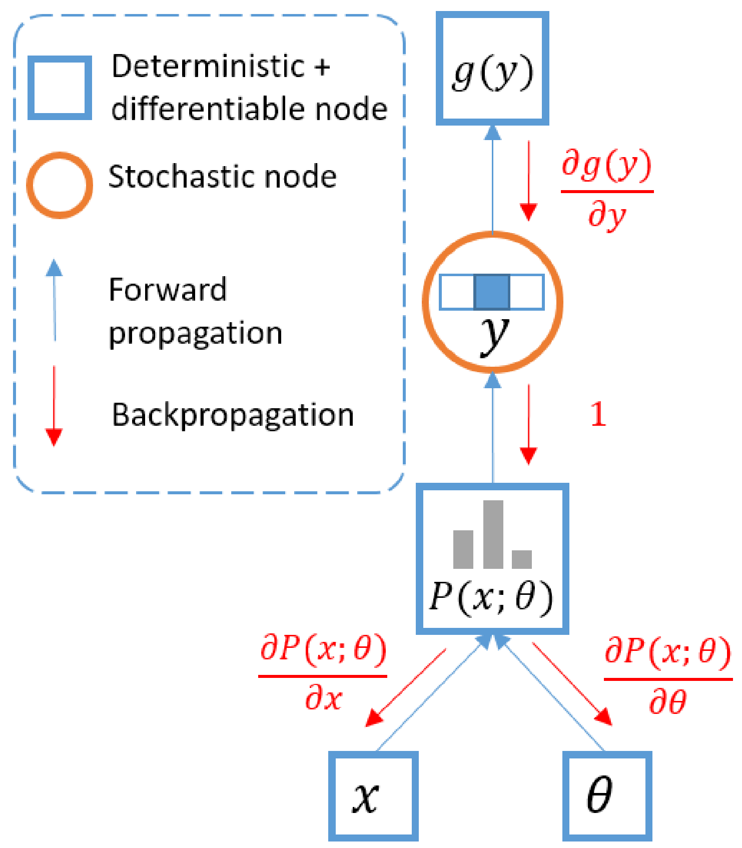
Feedback loss: $\mathcal{L}_{TTS} = ||\, x - \hat{x}||_2^2$ where $x = TTS(\hat{y}, z)$

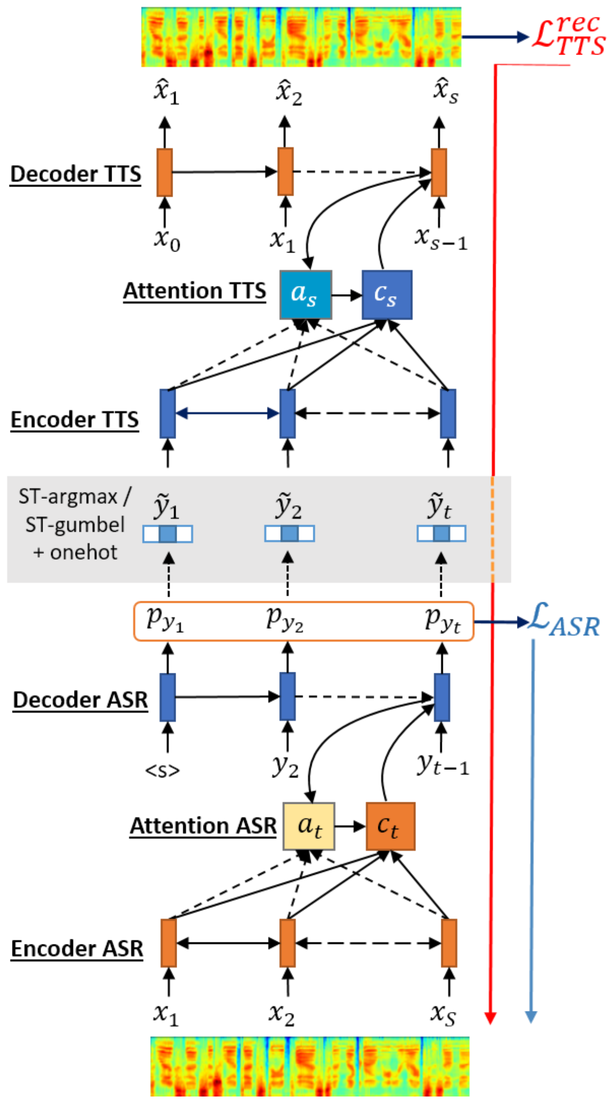
a) Speech chain loop with speaker embedding module.

b) Original: feedback $\mathcal{L}_{TTS}$ can't be backpropagated through variable $\hat{y}$.

c) **Proposal**: Estimate gradient through variable $\hat{y}$ with straight-through estimator.

3. Straight-through Estimator (ST)


Deterministic + differentiable node
Stochastic node
Forward propagation
Backpropagation
$g(y)$
$\frac{\partial g(y)}{\partial y}$
y
1
$P(x;\theta)$
$\frac{\partial P(x;\theta)}{\partial x}$
$\frac{\partial P(x;\theta)}{\partial \theta}$
x
θ



$$p_{y_t}[\mathrm{c}] = \frac{\exp\left(h_t^d[c]/\tau\right)}{\sum_{i=1}^{C} \exp\left(h_t^d\,[c]/\tau\right)}$$

τ = temperature

a) ST-argmax

Deterministic choosing token by highest probability.

$$\tilde{y}_t = argmax_c\; p_{y_t}[c]$$

b) ST-Gumbel softmax

Sampling a token from $p_{y_t}[c]$:

$$p_{y_t}[c] = \frac{\exp\left((h_t^d[c] + g_c)/\tau\right)}{\sum_{i=1}^{C} \exp\left((h_t^d[c] + g_c)/\tau\right)}$$

$$\tilde{y}_t \sim Cat\left(p_{y_t}[1], \ldots, p_{y_t}[C]\right)$$

New gradient $\mathcal{L}_{TTS}$ w.r.t. θ_{ASR}

$$\frac{\partial \mathcal{L}_{TTS}^{rec}}{\partial \theta_{ASR}} = \sum_{t=1}^{T} \frac{\partial \mathcal{L}_{TTS}^{rec}}{\partial \tilde{y}_t} \cdot \boxed{\frac{\partial \tilde{y}_t}{\partial p_{y_t}}} \cdot \frac{\partial p_{y_t}}{\partial \theta_{ASR}}$$

$$\approx \sum_{t=1}^{T} \frac{\partial \mathcal{L}_{TTS}^{rec}}{\partial \tilde{y}_t} \cdot \boxed{\mathbb{1}} \cdot \frac{\partial p_{y_t}}{\partial \theta_{ASR}}.$$

4. Experiment

- Features: log Mel-spec (80-dim)
- Text: 26 letters (A-Z)+(',-) + <noise>
- Dataset: Wall Street Journal
  - Train: train_si284 (WSJ1)
  - Dev: dev93 & Test: eval92
- Hyperparams $\tau = [0.25, 0.5, 1, 2]$

Result on WSJ-1

<table>
<tr><th colspan="4">Baseline</th></tr>
<tr><th colspan="3">Model</th><th>CER (%)</th></tr>
<tr><td colspan="3">Att MLP</td><td>7.12</td></tr>
<tr><td colspan="3">Att MLP-MA</td><td>6.43</td></tr>
<tr><th colspan="4">Proposed</th></tr>
<tr><th>Model</th><th>Generation</th><th>ST</th><th>CER (%)</th></tr>
<tr><td>Att MLP-MA</td><td>Teacher-forcing</td><td>argmax</td><td>5.75</td></tr>
<tr><td>Att MLP-MA</td><td>Teacher-forcing</td><td>gumbel</td><td>5.7</td></tr>
<tr><td>Att MLP-MA</td><td>Greedy</td><td>argmax</td><td>5.84</td></tr>
<tr><td>Att MLP-MA</td><td>Greedy</td><td>gumbel</td><td>5.88</td></tr>
</table>

5. Discussion

- We introduced ST-estimator for training ASR module based on TTS feedback loss.
- The gradient in discretization problem can be replaced by identity Jacobian matrix.
- Our experiment shows that by adding auxiliary feedback loss, we improve the ASR performance further.