# IMPROVING OOV DETECTION AND RESOLUTION WITH EXTERNAL LANGUAGE MODELS IN ACOUSTIC-TO-WORD ASR

Hirofumi Inaguma Masato Mimura Shinsuke Sakai Tatsuya Kawahara
Graduate School of Informatics, Kyoto University, Japan

## Background

**Acoustic-to-word end-to-end ASR**

Pros
- ◆ Extremely simplified architecture / training and decoding pipelines
- ◆ Fast decoding (applicable for the real time usage)
- ◆ Extract word-level representations ➡ dialogue, keyword spotting

Cons
- ◆ Data sparseness due to infrequent words
- ◆ Fixed word entry ➡ out-of-vocabulary (OOV) problem

- ➢ Pre-training with a phoneme-level model [Audhkhasi 2017]
- ➢ Multi-task learning (MTL) with an auxiliary character-level ASR (A2C) task [Li 2017, Ueno2018]
- ➢ OOV tokens are further recovered from character-level hypothesis [Li 2017, Ueno2018]

➡ A2W models are now open-vocabulary (at least)

## Problem of A2W ASR

- ➢ OOV detection is difficult
  - ➡ A2W is more likely to recognize OOV words (often infrequent words) incorrectly as other words in the predefined vocabulary
  - ➡ Confused with in-vocabulary words with similar pronunciation
  - ➡ They cannot be recovered by the A2C model
- ➢ Infrequent words should be recognized by the A2C model
  - ➡ A2C is more flexible than A2W for recognizing rare words
- ➢ How to detect OOV words accurately...
  - ➡ External LM trained with a large text has a role to detect them?

## Proposed method

### ◆ External LM integration for OOV detection (infrequent word recognition)

- ➢ Restrict the external LM vocabulary to that of the A2W model
- ➢ External LM has the better ability to detect OOV words based on contextual information since it is trained with a large-scale text
- ➢ Probability of the <OOV> class is boosted and OOV words get easier to be detected during inference
  - ➡ Increase the number of <OOV> tokens in the hypothesis
  - ➡ These <OOV> tokens are recovered by the A2C model

## System overview

### ◆ MTL with an auxiliary character-level ASR task

$$\mathcal{L}(x, y^{\mathrm{w}}, y^{\mathrm{c}}; \theta^{\mathrm{w}}, \theta^{\mathrm{c}}) = -\lambda \log P(y^{\mathrm{w}}|x) - (1-\lambda)\log P(y^{\mathrm{c}}|x)$$

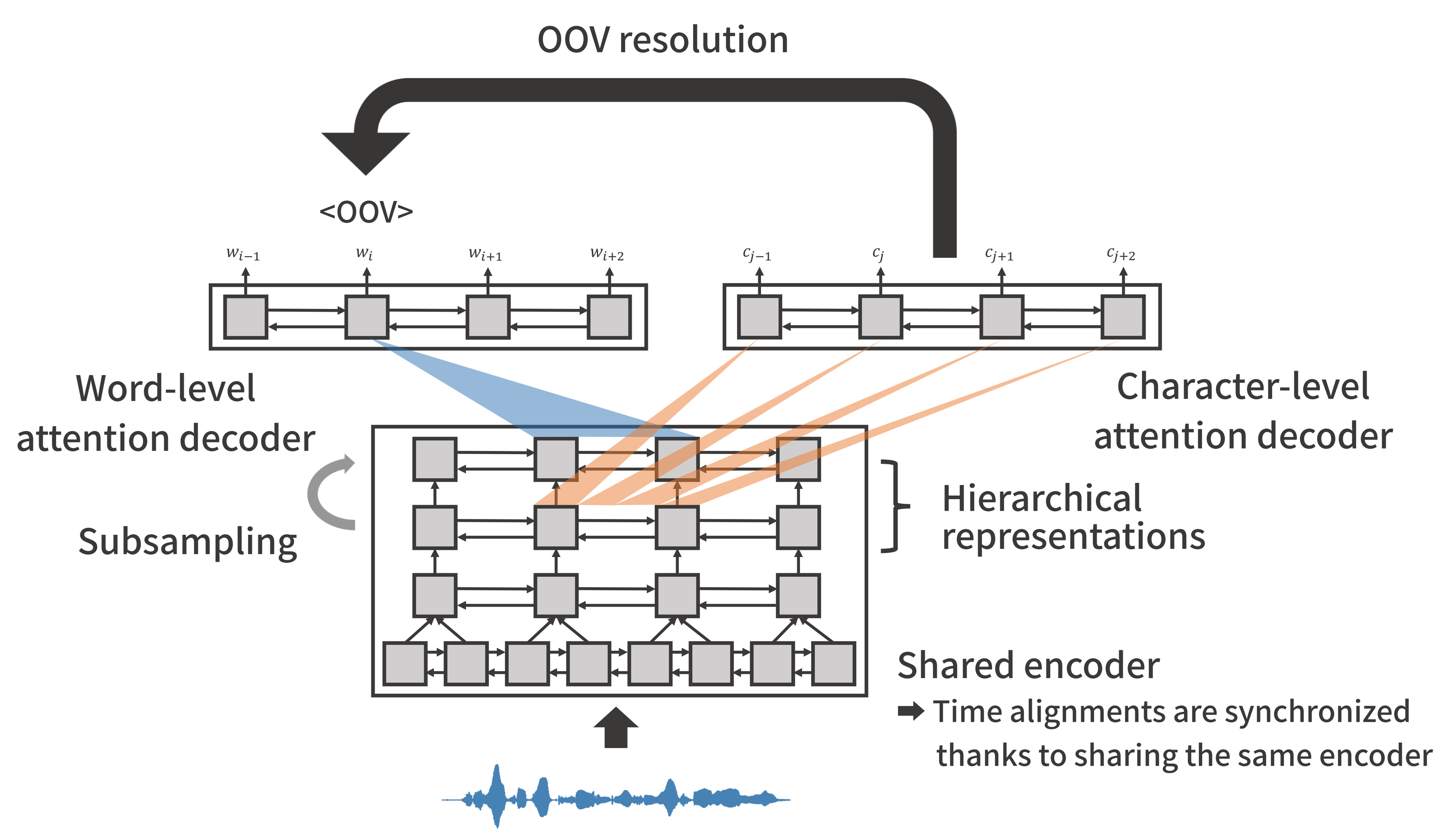
$\lambda$ : tunable parameter $(0 \le \lambda \le 1)$

### ◆ OOV resolution by character-level hypothesis

- ➢ Replace <OOV> tokens with the corresponding character from the A2C model by computing a position where attention distributions are most overlapped between the A2W and A2C models

### ◆ Word-level RNNLM integration (shallow fusion)

$$\widehat{y^{\mathrm{w}}} = \mathrm{argmax}_{y^{\mathrm{w}}} \{\log P_{\mathrm{A2W}}(y^{\mathrm{w}}|x) + \beta \log P_{\mathrm{WLM}}(y^{\mathrm{w}}) + \gamma\ coverage\}$$




## Experimental Evaluations

### ◆ Corpus

1. Switchboard
   - ASR: 300h
   - RNNLM: 2000h (+ Fisher)
2. CSJ (Japanese lecture corpus)
   - ASR: 240h (APS)
   - RNNLM: 600h (+ SPS)

### ◆ Architecture

- ✓ A2W: 5-layer BLSTM encoder + 1-layer LSTM decoder (320 memory cells)
- ✓ A2C: 4-layer BLSTM encoder + 1-layer LSTM decoder
- ✓ RNNLM: 2-layer LSTM (512 memory cells)
- ✓ $\lambda$=0.5, $\beta$=0.3, $\gamma$=0.6/0.2
- ✓ Beam width: 5 (A2W), 1 (A2C)

### ◆ Results on Switchboard

| Model | Resolving OOV | RNNLM | WER (#OOV) SWB | CH | Ave. |
|---|---|---|---|---|---|
| Word CTC | - | × | 20.26 (240) | 42.32 (358) | 31.29 |
| A2W (baseline) | - | × | 18.99 (154) | 38.46 (222) | 28.73 |
| | | 300h | 18.45 (319) | 38.13 (463) | 28.47 |
| | | 2000h | 18.35 (322) | 38.13 (490) | 28.24 |
| A2W + A2C (MTL) | × | × | 18.35 (183) | 37.54 (267) | 27.95 |
| | ○ | × | 18.18 ( 〃 ) | 37.40 ( 〃 ) | 27.79 |
| | × | 300h | 17.76 (349) | 37.26 (513) | 27.51 |
| | ○ | 300h | 17.43 ( 〃 ) | 36.99 ( 〃 ) | 27.21 |
| | × | 2000h | 17.40 (346) | 37.00 (546) | 27.20 |
| | ○ | 2000h | **17.11 ( 〃 )** | **36.71 ( 〃 )** | **26.91** |

4.7%

### ◆ Results on CSJ

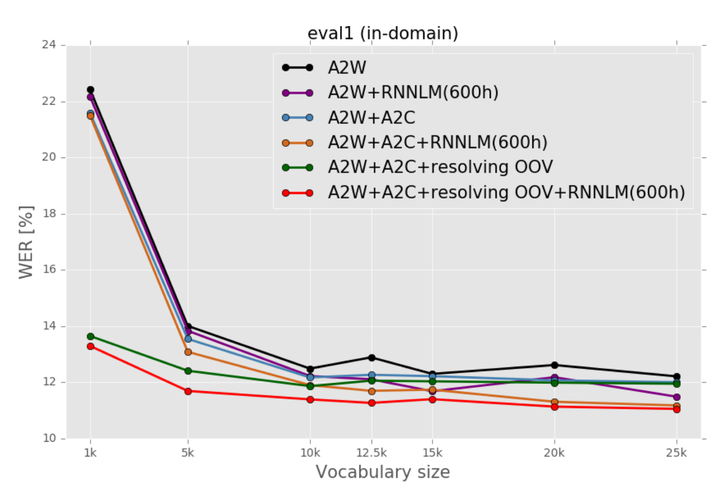
| Model | Resolving OOV | RNNLM | WER (#OOV) eval1 | eval2 | eval3* | Ave. |
|---|---|---|---|---|---|---|
| Word CTC | - | × | 12.79 (352) | 11.12 (469) | 20.28 (662) | 14.73 |
| A2W (baseline) | - | × | 12.89 (265) | 10.25 (299) | 19.70 (498) | 14.28 |
| | - | 240h | 12.20 (437) | 9.73 (531) | 19.49 (761) | 13.80 |
| | - | 600h | 12.11 (443) | 9.65 (516) | 18.71 (759) | 13.49 |
| A2W + A2C (MTL) | × | × | 12.27 (252) | 9.96 (334) | 18.70 (521) | 13.64 |
| | ○ | × | 12.06 ( 〃 ) | 9.67 ( 〃 ) | 17.99 ( 〃 ) | 13.24 |
| | × | 240h | 11.71 (441) | 9.40 (534) | 18.21 (782) | 13.11 |
| | ○ | 240h | 11.27 ( 〃 ) | 8.85 ( 〃 ) | 17.20 ( 〃 ) | 12.44 |
| | × | 600h | 11.70 (429) | 9.29 (518) | 17.54 (788) | 12.85 |
| | ○ | 600h | **11.27 ( 〃 )** | **8.77 ( 〃 )** | **16.57 ( 〃 )** | **12.21** |

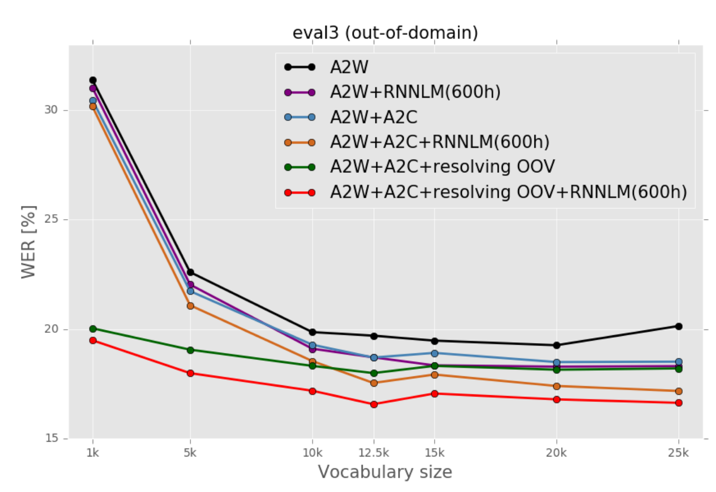
9.4%

* eval3 is the out-of-domain set

- ➢ External RNNLM increases the number of recognized <OOV> words
- ➢ MTL with an A2C model improves WER
- ➢ MTL enhances the effectiveness of RNNLM thanks to generalization effects
- ➢ Recovering <OOV> words by the A2C model further improves WER
- ➢ MTL + RNNLM integration + OOV resolution was the best
- ➢ Effective especially for the out-of-domain sets

### ◆ Analysis of the vocabulary size (CSJ)




- ➢ MTL + OOV resolution is robust to the vocabulary size
- ➢ MTL + RNNLM integration + OOV resolution is always effective

### ◆ Decoding speed (CSJ)

- ➢ With a single NVIDIA Titan GPU
- ➢ RTF is small enough for the real-time usage
- ➢ Computational cost
  - Small vocabulary: OOV resolution > RNNLM
  - Large vocabulary: OOV resolution < RNNLM
- ➢ A2W is faster than A2C