Cross-modal learning for CTC-based ASR: Leveraging CTC-BERTScore and sequence-level training

Mun-Hak Lee, Sang-Eon Lee, Ji-Eun Choi, Joon-Hyuk Chang

ASML Lab., Department of Electronic Engineering
Hanyang University, Seoul, Republic of Korea

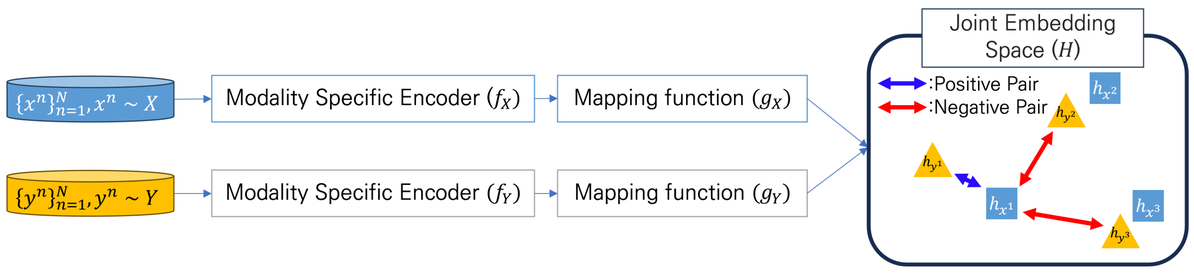
INTRODUCTION

- **Automatic Speech recognition (ASR)**
  1. ASR aims to determine the most probable token sequence Y inherent to given speech X.
  2. Decoding process of ASR model is expressed as:
  $$\hat{Y} = \text{argmax}_Y P(Y|X)$$
  3. To build a high-performance ASR model, $P(Y|X)$ must be approximated accurately.
- **Connectionist Temporal Classification (CTC)[1]**
  1. CTC is trained to maximize the likelihood of all possible alignments (a) a between token sequence Y and acoustic model (AM) output.
  $$P(Y|X) = \sum_{a \in \beta^{-1}(Y)} P(a|X)$$
  2. where β denotes a function that removes silence and repetitions.
  3. Based on the assumption of conditional independence, $P(a|X)$ can be approximated as follows:
  $$P(a|X) = \prod_{t=1}^{|a|} P(a_t|X, a_{0:t-1})\, P(a_0) \approx \prod_{t=1}^{|a|} P(a_t|X)$$
  3. where a_t denotes the token of time step t in alignment a, $a_{0:t-1}$ denotes all previous tokens before time t, a_0 denotes the start of the sentence token, and $|a|$ denotes the length of alignment a.
- **Limitation of CTC-based ASR**
  - ✓ The conditional independence assumption allows the modeling of $P(a|X)$ with a limited amount of computation; however, it is an incomplete assumption in that it does not model the correlation between tokens.

[1] Graves, Alex, and Alex Graves. "Connectionist temporal classification." Supervised sequence labelling with recurrent neural networks (2012): 61-93.

Background

- **Cross Modal Learning Paradigm**
  - Given two instances (x and y) instantiated in two different modalities, cross-modal learning aims to learn the correlations between instances with different modalities using a neural network model.
  - Common cross-modal learning methods consist of the following four steps:
    1. A representation vector $h_X = g_X(f_X(x))$ is generated on a fine-grained latent space with dimension d_H using a modality-specific encoder $f_X : X \mapsto \mathbb{R}^{d_X}$ and mapping function $g_X: \mathbb{R}^{d_X} \mapsto \mathbb{R}^{d_H}$.
    2. $h_Y = g_Y(f_Y(y))$ is generated in the same way using a modality-specific encoder $f_Y: Y \mapsto \mathbb{R}^{d_Y}$ and a mapping function $g_Y: \mathbb{R}^{d_Y} \mapsto \mathbb{R}^{d_H}$.
    3. The similarity function $S(\cdot,\cdot)$ is used to estimate the similarity between the two representation vectors ($h_X, h_Y \in \mathbb{R}^{d_H}$).
    4. The neural network is trained to increase the similarity between positive pairs and reduce the similarity between negative pairs with proper objective function.



〈 Schematic diagram of cross modal learning framework 〉

- **Cross Modal Learning from the Sequence Level Training Perspective**
  - Using the Bayes rule, $P(Y = y|X = x)$ can be approximated as follows:

$$P(Y = y|X = x) = \frac{P(X = x|Y = y)P(Y = y)}{P(X = x)}$$
$$= \frac{P(X = x|Y = y)P(Y = y)}{\sum_Y P(X = x|Y)P(Y)}$$

Sequence discriminative training such as: MMI, MBR[2,3]

$$\approx \frac{P(X = x|Y = y)P(Y = y)}{\sum_{\dot{y} \in \dot{y}} P(X = x|Y = \dot{y})P(Y = \dot{y})}$$
$$= \frac{P(X = x, Y = y)}{\sum_{\dot{y} \in \dot{y}} P(X = x, Y = \dot{y})}$$

  - where $\dot{y} = \{\dot{y}_1, \ldots \dot{y}_M\}$ denotes hypothesis set.
  - We estimate $P(X = x, Y = y)$ with neural networks following a cross modal learning scenario.

$$\frac{P(X = x, Y = y)}{\sum_{\dot{y} \in \dot{y}} P(X = x, Y = \dot{y})} = \frac{S(g_X(f_X(x)), g_Y(f_Y(y)))}{\sum_{\dot{y} \in \dot{y}} S(g_X(f_X(x)), g_Y(f_Y(\dot{y})))}$$

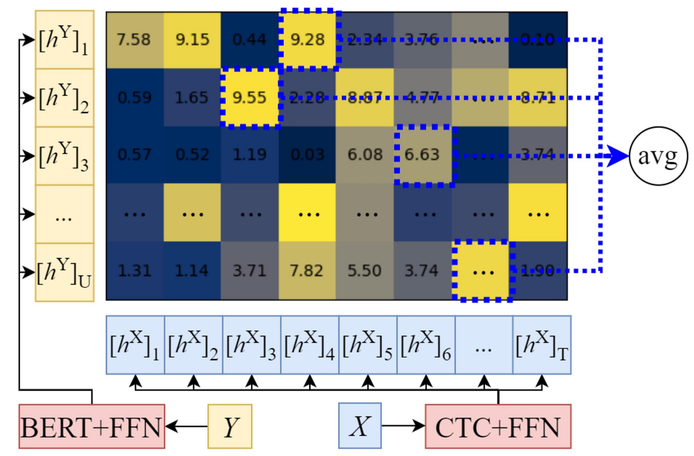
  - where $S(\cdot,\cdot)$ denotes similarity function $f_X(\cdot)$ denotes speech encoder, $f_Y(\cdot)$ denotes text encoder, and $g(\cdot)$ denotes trainable mapping function.
  - In the above equation, the numerator term represents the similarity between positive pairs, and the denominator term represents the similarity between negative pairs.

Reference (y)	Hypotheses ($\dot{y}_m$)	d_m	ψ_m	p_m^{ψ}
I love a dog	I love a dog	0	1	0.6103
	I love a a a a a dog	4	0.1353	0.0826
	I a dog	1	0.3679	0.2245
	I love dog a	2	0.1353	0.0826

〈 Example of sentence-level similarity distribution generation process based on edit-distance.〉

[2] Povey, Daniel. Discriminative training for large vocabulary speech recognition. Diss. University of Cambridge, 2005.
[3] Veselý, Karel, et al. "Sequence-discriminative training of deep neural networks." Interspeech. Vol. 2013. 2013.

Proposed Method



〈 Schematic diagram of the CTC-BERTScore measurement process 〉

- **CTC-BERT Score[4,5]**
  - CTC-BERTScore is a similarity measure that uses CTC and BERT models as modality specific encoders.
  - For a paired speech and text sample $(x, y) \sim \mathcal{P}(X, Y)$, let $h_X = g_X(f_{CTC}(X))$ and $h_Y = g_Y(f_{BERT}(Y; l_{BERT}))$. Where $f_{CTC}: X \mapsto \mathbb{R}^{T \times d_{CTC}}$ denotes the CTC model, $f_{BERT}: Y \mapsto \mathbb{R}^{U \times d_{BERT}}$ denotes the BERT, $g_X : \mathbb{R}^{T \times d_{CTC}} \mapsto \mathbb{R}^{T \times d_H}$ and $g_Y : \mathbb{R}^{U \times d_{BERT}} \mapsto \mathbb{R}^{U \times d_H}$ denote trainable mapping functions, T indicates the length of speech and U indicates the length of the token sequence. The CTC-BERTScore (recall and precision) is defined as follows:

$$R_{C.B}(x, y) = \frac{1}{T} \sum_{[h^X]_i \in h^X} \max_{[h^Y]_j \in h^Y} \frac{[h^X]_i^T [h^Y]_j}{\|[h^X]_i\| \|[h^Y]_j\|}, P_{C.B}(x, y)$$
$$= \frac{1}{U} \sum_{[h^Y]_j \in h^Y} \max_{[h^X]_i \in h^X} \frac{[h^X]_i^T [h^Y]_j}{\|[h^X]_i\| \|[h^Y]_j\|}$$

  - where $[h^X]_i$ denotes a representation vector of the i th time step in h^X, T denotes the length of speech, and U denotes the length of text.
- **Edit Distance-based Similarity**
  - The edit-distance-based similarity between the ith sentence $\dot{y}_i$ of the hypothesis set $\dot{y}$ and the reference sentence y can be defined as follows:

$$\psi_i = \exp\left(-\frac{d_i}{\tau \max(|y|, |\dot{y}_i|)}\right)$$

  - where $d_m = \text{Lev}(y, \dot{y}_m)$, $\text{Lev}(\cdot,\cdot)$ denotes Levenshtein distance, y denotes reference.
- **Cross-Modal Matching Word Error Distribution (CMWED)[6]**
  - CMWED is a auxiliary sequence level objective function for CTC model training.
  - Given the edit-distance based sequence-level similarity ψ_m and CTC-BERTScore ($P_{C.B}(x, \dot{y}_m)$ or $R_{C.B}(x, \dot{y}_m)$) for the mth hypothesis $\dot{y}_m$ in the hypothesis set, the CMWED loss is defined as follows:

$$\mathcal{L}_{CMWED} = \sum_{m=1}^{M} -p_m^{\psi} \log p_m^{P_{C.B}},$$
$$p_m^{\psi} = \frac{\psi_m}{\sum_{i=1}^{M} \psi_i}, p_m^{P_{C.B}} = \frac{P_{C.B}(x, \dot{y}_m)}{\sum_{i=1}^{M} P_{C.B}(x, \dot{y}_i)}$$

  - The final loss function is expressed as follows:
  $$\mathcal{L}_{Total} = \mathcal{L}_{CTC} + \alpha \mathcal{L}_{CMWED}$$
  - where $\alpha \geq 0$ is a hyperparameter.

[4] Kenton, Jacob Devlin Ming-Wei Chang, and Lee Kristina Toutanova. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding." Proceedings of NAACL-HLT. 2019.
[5] Zhang, Tianyi, et al. "BERTScore: Evaluating Text Generation with BERT." International Conference on Learning Representations. 2019.
[6] Liyan Xu, Yile Gu, Jari Kolehmainen, Haidar Khan, Ankur Gandhe, Ariya Rastrow, Andreas Stolcke, and Ivan Bulyko, "Rescorebert: Discriminative Speech Recognition Rescoring with BERT," in Proc. of 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6117–6121.

Experiments

- We finetuned the trained Wav2Vec 2.0[7] base model and used the LibriSpeech dataset.

Table 1. WER results for CTC models trained with the LibriSpeech100h/960h dataset and CMWED loss. l_{BERT} denotes the number of transformer layers used to create the BERT representation vector, and $S(\cdot,\cdot)$ denotes the similarity measure and $\dot{y}$ generation refers to the method used to create the hypothesis set.

l_{BERT}	α	$S(\cdot,\cdot)$	$\dot{y}$ generation	Trainset (h)	WER (%, w/o LM)				WER (%, 4-gram LM)			
					test		dev		test		dev	
					clean	other	clean	other	clean	other	clean	other
-	0	-	-	100	6.13	12.95	6.06	13.43	3.44	8.24	2.79	8.07
12	$1/U$	$P_{\text{C.B}}$	Aug	100	5.68	13.02	5.61	13.38	3.32	8.18	2.76	8.02
12	$1/T$	$R_{\text{C.B}}$	Aug	100	5.67	**12.76**	5.66	13.47	3.31	**8.05**	2.71	8.08
12	$1/T$	$R_{\text{C.B}}$	M-best	100	**5.52**	13.00	**5.55**	13.51	**3.25**	8.26	**2.66**	**7.99**
9	$1/T$	$R_{\text{C.B}}$	Aug	100	5.82	12.85	5.72	13.47	3.32	8.17	2.73	8.01
7	$1/T$	$R_{\text{C.B}}$	Aug	100	5.65	12.98	5.66	13.50	3.37	8.32	2.70	8.01
3	$1/T$	$R_{\text{C.B}}$	Aug	100	5.71	12.83	5.64	**13.34**	3.31	8.17	2.68	8.07
1	$1/T$	$R_{\text{C.B}}$	Aug	100	5.79	12.90	5.65	**13.34**	3.38	8.23	2.81	8.15
12	$0.1/T$	$R_{\text{C.B}}$	Aug	100	5.63	13.04	5.72	13.49	3.29	8.21	2.72	8.15
12	$10/T$	$R_{\text{C.B}}$	Aug	100	5.71	13.22	5.71	13.74	3.34	8.35	2.76	8.33
-	0	-	-	960	3.34	8.61	3.14	8.82	2.65	6.15	1.91	5.81
12	$1/T$	$R_{\text{C.B}}$	M-best	960	3.20	8.45	3.13	8.59	2.56	6.02	1.88	5.66

- We experimentally confirmed that the proposed method improves the generalization performance of the CTC model, and claimed that the representation power of the BERT model, which is good at embedding contextual information, was transferred to the CTC model, resulting in performance improvement.

[7] Baevski, Alexei, et al. "wav2vec 2.0: A framework for self-supervised learning of speech representations." Advances in neural information processing systems 33 (2020): 12449-12460.