Interspeech 2022
18-22 September 2022, Incheon, Korea

Knowledge Transfer and Distillation from Autoregressive to Non-Autoregressive Speech Recognition

Xun Gong, Zhikai Zhou, Yanmin Qian[†]

MoE Key Lab of Artificial Intelligence, AI Institute
X-LANCE Lab, Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China
{gongxun,zhikai.zhou,yanminqian}@sjtu.edu.cn

Abstract

Modern non-autoregressive (NAR) speech recognition systems aim to accelerate the inference speed; however, they suffer from performance degradation compared with autoregressive (AR) models as well as the huge model size issue. We propose a novel knowledge transfer and distillation architecture that leverages knowledge from AR models to improve the NAR performance while reducing the model's size. Frame- and sequence-level objectives are well-designed for transfer learning. To further boost the performance of NAR, a beam search method on Mask-CTC is developed to enlarge the search space during the inference stage. Experiments show that the proposed NAR beam search relatively reduces CER by over 5% on AISHELL-1 benchmark with a tolerable real-time-factor (RTF) increment. By knowledge transfer, the NAR student who has the same size as the AR teacher obtains relative CER reductions of 8/16% on AISHELL-1 dev/test sets, and over 25% relative WER reductions on LibriSpeech test-clean/other sets. Moreover, the ~9x smaller NAR models achieve ~25% relative CER/WER reductions on both AISHELL-1 and LibriSpeech benchmarks with the proposed knowledge transfer and distillation.

Index Terms: knowledge transfer, knowledge distillation, non-autoregressive, end-to-end, speech recognition

1. Introduction

In recent years, the performance of automatic speech recognition (ASR) has been greatly improved by sequence-to-sequence modeling, such as connectionist temporal classification (CTC) [1], recurrent neural network transducer (RNN-T) [2], and attention-based encoder-decoder (AED) [3]. Many of the earlier researches have focused on autoregressive (AR) modeling, which generates the token sequence using a left-to-right probabilistic chain rule. Despite their great performance, such AR models require L-step incremental model calculations to generate L tokens, resulting in high inference latency and considerable computational costs.

From another aspect, non-autoregressive (NAR) modeling generates the token sequence in constant steps, and removes the chain-rule assumption. CTC [1] plays an essential role in recent NAR researches. Modern NAR approaches outperform CTC by leveraging alignments (align-based) and the output token sequence (token-based). Based on the joint CTC/attention architecture [3], Mask-CTC [4] utilizes the (conditional) masked language model ((C)MLM) decoder to refine the CTC token sequence. [5] proposes two auxiliary tasks to solve the length prediction problem that occurred in Mask-CTC. From another point of view, CTC alignment shows its advantage for building NAR models in Align-Refine [6], CASS-NAT [7] and AL-NAT [8]. Moreover, the self-supervised model wav2vec2.0 [9] has achieved promising results by CTC.

However, there are two major challenges remaining for NAR modeling: First, NAR models converge slowly and perform poorly compared to state-of-the-art (SOTA) AR models [4, 10]. Second, although NAR models are often favored in resource-constrained situations for fast inference speed and high accuracy [4], the large model scale and computational cost limit the application of NAR modeling. Knowledge distillation (Transfer learning) is typically used to solve such a problem by teaching a smaller student model [11]. To be specific, the student aims to mimic the soft targets provided by the well-trained teacher using the Kullback-Leibler divergence (KLD) [12, 13, 14]. Nevertheless, when applying knowledge distillation on non-autoregressive ASR, the poor NAR teacher limits the improvement.

In this paper, we propose a novel architecture by transferring and distilling the knowledge from an autoregressive (AR) teacher to a non-autoregressive (NAR) student together with a beam-search decoding method to boost the performance of non-autoregressive modeling. Firstly, we introduce a beam-search decoding method to enlarge the search space for the (conditional) masked language model ((C)MLM) decoder [15]. Then, we extend the knowledge distillation technique by transferring the AR teacher's knowledge to NAR in two distillation levels, therefore improving NAR students' performance. The encoder distillation is conducted following our previous setup [12]. For the decoder distillation, we develop the frame- and sequence-level distillation from the attention-based autoregressive model into Mask-CTC. The distillation loss is customized for token-based NAR models, so that the NAR decoder can benefit from the AR decoder.

The structure of the paper is organized as follows: In Section 2, the attention-based autoregressive model and non-autoregressive Mask-CTC are briefly introduced. In Section 3, we present the proposed beam search method for Mask-CTC and the knowledge distillation from AR to NAR ASR. In Section 4, experimental results and analysis are given on the Mandarin AISHELL-1 and English LibriSpeech datasets. Finally, the conclusion is drawn in Section 5.

2. Autoregressive and Non-autoregressive ASR

Basically, end-to-end ASR models map speech features $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_T]^\intercal, \boldsymbol{x}_t \in \mathbb{R}^F$ to a token sequence $\boldsymbol{y} = [y_1, y_2, \cdots, y_L]^\intercal, y_l \in \mathcal{U}$, where F is the feature dimension

[†] corresponding author

10.21437/Interspeech.2022-632

and $\mathcal{U}$ denotes the vocabulary set.

Traditional attention-based autoregressive (AR) ASR models [16, 17] firstly encode speech features $\boldsymbol{X}$ into a hidden representation $\boldsymbol{H}$: $\boldsymbol{H} = \text{Encoder}(\boldsymbol{X})$, and then compose it with previous tokens $\boldsymbol{y}_{<l}$ to estimate the posterior $p(y_l|\boldsymbol{y}_{<l}, X)$:

$$p_{\text{ar}}(y_l|\boldsymbol{y}_{<l}, \boldsymbol{H}) = \text{Decoder}(\boldsymbol{y}_{<l}, \boldsymbol{H}), \tag{1}$$

and the whole sequence probability is

$$p_{\text{ar}}(\boldsymbol{y}|\boldsymbol{H}) = p_{\text{ar}}(y_1|\boldsymbol{H}) \prod_{l=2}^{L} p_{\text{ar}}(y_l|\boldsymbol{y}_{<l}, \boldsymbol{H}). \tag{2}$$

During inference, the AR model generates hypothesis $\hat{\boldsymbol{y}}$ token-by-token.

Connectionist temporal classification (CTC) [1] is one of the earliest non-autoregressive (NAR) method, which introduces a many-to-one function η from the frame-level alignment $\boldsymbol{Z} = [z_1, z_2, \cdots, z_T]^\intercal, z_t \in \mathcal{U} \cup \{\texttt{blank}\}$ to the token sequence $\boldsymbol{y}$, by merging same labels and removing the `blank` in $\boldsymbol{Z}$. The sequence probability is represented as:

$$p_{\text{ctc}}(\boldsymbol{y}|\boldsymbol{H}) = \sum_{\boldsymbol{Z} \in \eta^{-1}(\boldsymbol{y})} \prod_{t} p_{\text{ctc}}(z_t|\boldsymbol{H}), \tag{3}$$

where η is a many-to-one function from $\boldsymbol{Z}$ to $\boldsymbol{y}$. During inference, greedy CTC predicts the alignment by selecting the tokens with the highest probability for each step.

Mask-CTC [4], which is a popular instantiation of NAR ASR, is actually a refinement of CTC results via the conditional masked language model (MLM) [15]. During training, the groundtruth $\boldsymbol{y}$ are randomly replaced by the special token <MASK> , and the MLM decoder predicts masked tokens $\boldsymbol{y}_{\text{mask}}$, conditioned on the observed tokens $\boldsymbol{y}_{\text{obs}} = \boldsymbol{y} \setminus \boldsymbol{y}_{\text{mask}}$:

$$p_{\text{mlm}}(\boldsymbol{y}_{\text{mask}}|\boldsymbol{y}_{\text{obs}}, \boldsymbol{H}) = \prod_{y \in \boldsymbol{y}_{\text{mask}}} p_{\text{mlm}}(y|\boldsymbol{y}_{\text{obs}}, \boldsymbol{H}). \tag{4}$$

During inference, the output is initialized by CTC greedy decoding, and low-confidence tokens are substituted by <MASK> based on pre-defined threshold p_{thr}. After that, masks are filled using easy-first algorithm: fill all masks in $\lceil N/K \rceil$ iterations, where N denotes the total number of <MASK> and each iteration predicts top-k tokens with the highest confidence guided by MLM:

$$\hat{\boldsymbol{C}} = \underset{\substack{\boldsymbol{C} \subset \boldsymbol{y}_{\text{mask}}, \\ |\boldsymbol{C}|=k}}{\arg\max} \prod_{c \in \boldsymbol{C}} p_{\text{mlm}}(c|\boldsymbol{y}_{\text{obs}}, \boldsymbol{H}), \tag{5}$$

where $k = \begin{cases} N \mod K, & \text{the last iteration} \\ K, & \text{otherwise} \end{cases}$, $\boldsymbol{C}$ is the candidate set of <MASK> tokens and $\widehat{\boldsymbol{y}}'_{\text{obs}} = \boldsymbol{y}_{\text{obs}} + \hat{\boldsymbol{C}}$ is the updated result after mask filling.

The joint CTC/attention architecture [3, 4] is widely used in modern AR and NAR ASR models, with a multi-task learning based loss function:

$$\mathcal{L}_{\text{jca}} = \alpha \mathcal{L}_{\text{ctc}} + (1 - \alpha) \mathcal{L}_{\text{att}}, \tag{6}$$

where $\alpha \in [0, 1]$ is a hyperparameter, $\mathcal{L}_{\text{att}} = \mathcal{L}_{\text{ar}}$ for AR ASR, and $\mathcal{L}_{\text{att}} = \mathcal{L}_{\text{mlm}}$ for NAR ASR, respectively.

3. Proposed Methods

In this section, we introduce: (1) the proposed beam search method for NAR, (2) the distillation architecture transferring knowledge from AR to NAR ASR.

3.1. Beam Search for NAR ASR

Inspired by joint and rescoring decoding [3], we design a beam-search decoding method to enlarge the search space for the MLM decoder. The procedure is shown in Algorithm 1. Ω_1 is a sorted queue to be updated at the beginning of one iteration, Ω_0 stores the final Ω_1 after one iteration. During each iteration, a B-size beam is preserved, and the number of updated tokens is fixed and computed by k (i.e. $|\widehat{y}^{i+1}_{\text{obs}}| = |\widehat{y}^{i}_{\text{obs}}| + k$). Top-$B$ candidates are selected, according to the log domain posterior probability and Equation 5.

```
Algorithm 1: Beam Search on NAR MLM Decoding
$\hat{\boldsymbol{y}} \leftarrow$ greedy CTC ;
$\hat{\boldsymbol{y}}_{\text{mask}}$ and $\hat{\boldsymbol{y}}^0_{\text{obs}}$ by $p_{thr}$ ;
$\Omega_0 \leftarrow \{\hat{\boldsymbol{y}}^0_{\text{obs}}\}$: a list to store accepted hypotheses ;
for each iteration $i = 1, 2, \cdots, \lceil N/K \rceil$ do
    $\Omega_1 \leftarrow \varnothing$: $\Omega_1$ is a sorted queue with $B$ length ;
    $k = \begin{cases} N \mod K, & \text{the last iteration,} \\ K, & \text{otherwise.} \end{cases}$ ;
    for $\boldsymbol{y}_{obs} \in \Omega_0$ do
        Get top-$B$ candidate sets $\widehat{\boldsymbol{y}}'^{1}_{\text{obs}}, \cdots, \widehat{\boldsymbol{y}}'^{B}_{\text{obs}}$ by the
         posterior in Equation 5 ;
        Push $\widehat{\boldsymbol{y}}'^{1}_{\text{obs}}, \cdots, \widehat{\boldsymbol{y}}'^{B}_{\text{obs}}$ into $\Omega_1$ ;
    $\Omega_0 = \Omega_1$ ;
return $\arg\max_{\hat{\boldsymbol{y}} \in \Omega_0} \hat{\boldsymbol{y}}$ ;
```

3.2. Knowledge Transfer and Distillation from Autoregressive to Non-Autoregressive ASR

As previously stated, knowledge distillation performance on NAR is constrained owing to the poor performance of the NAR teacher. Here, we propose knowledge transfer and distillation from autoregressive (AR) to non-autoregressive (NAR) ASR, pushing the limits of NAR.

Firstly, we introduce two types of knowledge distillation techniques based on Kullback-Leibler divergence (KLD): $KLD(P, Q) = \sum_i P_i \log(P_i/Q_i)$, where P, Q are the teacher and student output distributions, respectively.

Frame-level knowledge distillation as a basic distillation criterion is formulated as below:

$$\mathcal{L}_{\text{F-KD}} = -\sum_{t=1}^{T} \sum_{c \in \mathcal{U}} P_t(c) \log Q_t(c), \tag{7}$$

where $P_t(c)$ and $Q_t(c)$ are the posterior probabilities of token c at timestamp t of teacher P and student Q. $\boldsymbol{H}$, $\boldsymbol{y}_{\text{obs}}$, and $\boldsymbol{y}_{<t}$ are the conditions of the above probabilities, but omitted for simplicity. $P_t(c) \log P_t(c)$ is omitted in computing KLD loss due to the frozen teacher during training.

Sequence-level knowledge distillation is another distillation criterion:

$$\mathcal{L}_{\text{S-KD}} = -\sum_{\widehat{\boldsymbol{y}}^i \in \mathcal{T}} P(\widehat{\boldsymbol{y}}^i) \log Q(\widehat{\boldsymbol{y}}^i) \tag{8}$$

where $\widehat{\boldsymbol{y}}^i$ is the hypotheses from teacher model, $\mathcal{T}$ is the set of all possible sequences and similar omission as in Equation 7. Using such sequence-level knowledge distillation is unaffordable, as we are approximating an exponentially-sized sequence

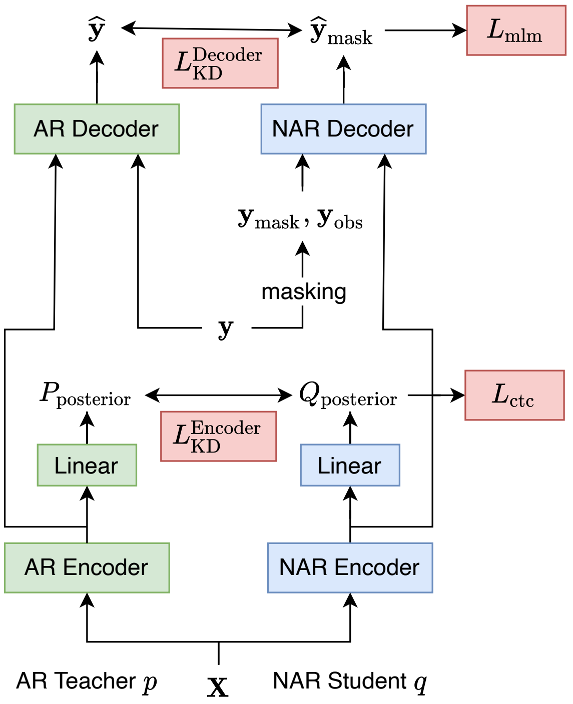

Figure 1: *Overview of proposed knowledge distillation from autoregressive to non-autoregressive ASR.*

distribution $\mathcal{T}$. Similar to MWER [18] training, N-best candidates set Ω is accessed by beam search, and then $P(\widehat{\boldsymbol{y}}^i)$ can be approximated by:

$$P(\widehat{\boldsymbol{y}}^i) \simeq \frac{P(\widehat{\boldsymbol{y}}^i)}{\sum_{\widehat{\boldsymbol{y}}^j \in \Omega} P(\widehat{\boldsymbol{y}}^j)}. \tag{9}$$

And then we can achieve the knowledge distillation loss by:

$$\mathcal{L}_{\text{KD}} = \beta_F \mathcal{L}_{\text{F-KD}} + \beta_S \mathcal{L}_{\text{S-KD}}, \tag{10}$$

where β_F, β_S are hyper-parameters for frame-level knowledge distillation loss $\mathcal{L}_{\text{F-KD}}$ and sequence-level $\mathcal{L}_{\text{S-KD}}$, respectively.

As shown in Figure 1, the proposed knowledge distillation methods are divided into two parts: the first part is the distillation after the encoder, and the second part is the distillation after the decoder. The encoder distillation $\mathcal{L}_{\text{KD}}^{\text{Encoder}}$ is done after the linear layer of the encoder, which has $\mathcal{L}_{\text{F-KD}}$ and $\mathcal{L}_{\text{S-KD}}$ similar to literature [12]. The decoder distillation is setup as follows. For frame-level distillation, only $\boldsymbol{y}_{\text{mask}}$ positions are selected, so the objective function is normalized by the number of <MASK> tokens:

$$\mathcal{L}_{\text{F-KD}}^{\text{Decoder}} = -\sum_{t \in \boldsymbol{y}_{\text{mask}}} \sum_{c \in \mathcal{U}} \frac{P_t(c|\boldsymbol{y}_{<t}, \boldsymbol{H})}{|\boldsymbol{y}_{\text{mask}}|} \log Q_t(c|\boldsymbol{y}_{\text{obs}}, \boldsymbol{H}). \tag{11}$$

For sequence-level distillation, approximate probability P' from N-best Ω is used:

$$\mathcal{L}_{\text{S-KD}}^{\text{Decoder}} = -\sum_{\hat{\boldsymbol{y}} \in \Omega} \frac{P'(\hat{\boldsymbol{y}})}{|\boldsymbol{y}_{\text{mask}}|} \log Q(\boldsymbol{y}_{\text{mask}}|\boldsymbol{y}_{\text{obs}}, \boldsymbol{H})). \tag{12}$$

And the final loss is then:

$$\mathcal{L} = \mathcal{L}_{\text{jca}}^{\text{student}} + \gamma_{\text{enc}} \mathcal{L}_{\text{KD}}^{\text{Encoder}} + \gamma_{\text{dec}} \mathcal{L}_{\text{KD}}^{\text{Decoder}}, \tag{13}$$

where $\gamma_{\text{enc}}, \gamma_{\text{dec}}$ are weight coefficients for encoder and decoder knowledge distillation.

4. Experiments

4.1. Datasets

Our experiments are conducted on the Mandarin AISHELL-1 [19] and the English LibriSpeech corpora [20]. AISHELL-1 contains a 150h training set, with a development (dev) and test set for evaluation, while LibriSpeech has a 960h training set, with test-clean/other (test c/o) used for tests. We report the character error rate (CER) on AISHELL-1, word error rate (WER) on LibriSpeech, respectively.

4.2. Model description

For acoustic feature extraction, 80-dimensional mel filterbank (Fbank) features are extracted with global level cepstral mean and variance normalization (CMVN). When it comes to data augmentation, speed perturbation is applied only for AISHELL-1 and SpecAugment [21] for both datasets, respectively. For text modeling, 5000 English Byte Pair encoding (BPE) [22] subword units are adopted for English, and 4233 characters for Mandarin. The baseline follows the recipe of ESPnet v2 [23], a 12-layer conformer encoder with four times down-sampling and a 6-layer transformer decoder. The weight α for CTC module is fixed to 0.3.

For knowledge transfer and distillation, we firstly train a new NAR student model from scratch with $\mathcal{L}_{\text{F-KD}}$ for 80 epochs. The hyper-parameters are set to $\beta_F = 1.0, \beta_S = 0, \gamma_{\text{enc}} = 0.5, \gamma_{\text{dec}} = 0.3$. Then we fine-tune the distillation procedure by adding $\mathcal{L}_{\text{S-KD}}$, and the tuning parameters are set as $\beta_F = 1.0, \beta_S = 1.0, \gamma_{\text{enc}} = 0.5, \gamma_{\text{dec}} = 0.5$ with 20 epochs. In Equation 8, 9, we use beam-size $|\Omega| = 10$, which is consistent with the decoding hyper-parameters in AR model.

Different NAR student model sizes are explored in Table 2, identified as large (L), medium (M), small (S), and extremely small (XS). The AR teacher model keeps L size for LibriSpeech, and M-size for AISHELL-1.

In the inference stage, no language model is used during the following experiments. Model parameters are averaged over the last 5 checkpoints. For autoregressive models, joint CTC/attention one-pass decoding [3] is used with beam size 10, and the score interpolation of CTC is 0.3. For non-autoregressive Mask-CTC decoding, we follow the beam decoding method in Section 3.1, with beam $B = 10$, the threshold $p_{\text{thr}} = 0.99$ and $K = 2$, for AISHELL-1 and LibriSpeech corpus.

4.3. Results with the NAR Beam Decoding

As proposed in Section 3.1, we firstly evaluate the beam search performance with real time factor (RTF) in Table 3. RTF is computed using Intel-Xeon E5-2690 CPU with a single core at test set. To be consistent with literature [4], the NAR (M) model speeds up AR (M) [3] model by more than 10x, as the RTF is 0.58 for AR (M) and 0.31 for AR (S). Without too much degradation of inference speed (1.5x as slow as 'Beam1'), the beam decoding method achieves better performance compared with the greedy (Beam1) one by 5%∼9% relative WER reduction on

Table 1: *Knowledge transfer and distillation performance on AISHELL1 corpus (CER) (%) and Librispeech test corpus (WER) (%), 'I+D' is the sum of insertion and deletion errors, and 'A' is the total CER/WER. '#Param' has brackets with 'XS', 'S', 'M', 'L', which are mentioned in Table 2. 'Same size' means NAR has the same model scale as AR, 'Smaller' means NAR is* ∼*9x smaller than AR.*

Method	Criterion	AISHELL1					Librispeech Test				
		#Param	Dev		Test		#Param	clean		other	
			I+D	A	I+D	A		I+D	A	I+D	A
W2V2-CTC [10, 9]	-	95M	-	4.8	-	5.3	95M	-	2.7	-	8.0
CASS-NAT v2 [7]	-	-	-	4.9	-	5.4	-	-	3.1	-	7.2
AL-NAT [8]	-	71.3M	-	4.9	-	5.3	∼115M	-	3.2	-	7.4
AR	$\mathcal{L}_{jca}$	46.8M (M)	0.2	4.5	0.2	4.9	115.0M (L)	0.2	2.8	1.3	6.6
NAR (Same size)	$\mathcal{L}_{jca}$	46.8M (M)	0.4	5.4	0.5	6.4	115.0M (L)	0.6	4.1	2.0	10.2
	+$\mathcal{L}_{\text{F-KD}}$	-	0.2	**5.1**	0.3	5.6	-	0.5	3.5	1.6	8.4
	++$\mathcal{L}_{\text{S-KD}}$	-	0.2	**5.0**	0.2	**5.4**	-	0.4	**3.3**	1.4	**7.8**
NAR (Smaller)	$\mathcal{L}_{jca}$	6.5M (XS)	0.4	7.2	2.1	8.5	12.4M (S)	0.8	5.1	2.4	12.4
	+$\mathcal{L}_{\text{F-KD}}$	-	0.2	**6.1**	0.5	6.9	-	0.4	3.8	1.8	10.3
	++$\mathcal{L}_{\text{S-KD}}$	-	0.2	**6.0**	0.3	**6.5**	-	0.3	**3.7**	1.4	**9.2**

Table 2: *Model hyper-parameters for different AR and NAR Conformer scales at L, M, X and XS.*

Model	L	M	S	XS
#Params (M)	115.0	46.8	12.4	6.5
Attention Dim	512	256	128	128
Attention Heads	8	4	4	4
Inner-dim	2048	2048	1024	256

the test set. As the beam size B grows, the rate of improvement decreases.

Table 3: *Non-autoregressive Mask-CTC Performance (CER) on AISHELL-1 corpus. Real time factor (RTF) are reported for the test set.*

Decoding	Dev (%)	Test (%)	RTF
Non-Autoregressive (M)			
Beam1	5.4	6.7	0.051
Beam5	5.4	6.4	0.064
Beam10	5.4	6.4	0.072
Non-Autoregressive (XS)			
Beam1	7.6	9.2	0.017
Beam5	7.3	8.6	0.023
Beam10	7.2	8.5	0.027

4.4. Results on Knowledge Transfer and Distillation

Table 1 compares the knowledge transfer distillation and other modern AR and NAR models on AISHELL-1 and LibriSpeech datasets to validate the performance.

AISHELL-1: As shown in Table 1, the teacher AR model obtains more than 24% relative reduction on CER compared with NAR (M), and 40% with NAR (XS). After knowledge distillation, the NAR (M) with '+$\mathcal{L}_{\text{F-KD}}$' achieves 8% and 16% relative CER reduction on dev and test sets respectively, and the one based on '+$\mathcal{L}_{\text{F-KD}}$' with '++$\mathcal{L}_{\text{S-KD}}$' shows a further CER reduction on test set by 15%. The student results achieve competitive performance (5.0%/5.4% CER) to the state-of-the-art NAR models like CASS-NAT [7] or AL-NAT [8]. Similar results are obtained for distilled NAR (XS) as 18%/25% CER reduction on two evaluation sets.

LibriSpeech: Table 1 shows the performance comparison on the large LibriSpeech corpus. The AR (L) is adopted as teacher model while NAR (L,S) as student model. The observations are consistent with that in Table 1, $\mathcal{L}_{\text{F-KD}}$ and $\mathcal{L}_{\text{S-KD}}$ further boost the performance of NAR Mask-CTC model at L (3.3/7.8% WER) and S (3.7/9.2% WER) scale by ∼25% relative WER reduction. However, due to the limits of the AR teacher, the insertion and deletion error rate is high on AR (L).

Results show that such our method narrows the gap between AR and NAR, with the improvement being significantly greater in the more difficult evaluation set (i.e. test set in AISHELL-1, test-other in LibriSpeech). After knowledge transfer and distillation, the length error problem is greatly alleviated compared with the original NAR model owing to the high prediction accuracy of AR teacher. Moreover, both $\mathcal{L}_{\text{F-KD}}$ and $\mathcal{L}_{\text{S-KD}}$ attribute to reducing insertion and deletion errors ('I+D'), pushing the length error problem [4] to its limits at 0.2% CER for 'I+D' in AISHELL-1 and 1.4% for LibriSpeech test-other. Meanwhile, the NAR student model performs comparable results with other state-of-the-art NAR methods, including wav2vec2-CTC [9, 10], Improved CASS-NAT [7] and AL-NAT [8].

5. Conclusions

In this paper we propose a novel knowledge transfer and distillation architecture that leverages knowledge from AR models to improve NAR performance while reducing the model size. To further boost the performance of NAR, we propose a beam search method on Mask-CTC, which enlarges the search space during inference stage. Experiments demonstrate that NAR beam search obtains relative 5% reduction in AISHELL-1 dataset with a tolerable RTF increment. For knowledge distillation, most results achieve over 15% relative CER/WER reduction on large and smaller NAR modeling. Future works are going to explore the generalization of knowledge distillation from AR to NAR. Different non-autoregressive like CASS-NAT [7] and AL-NAT [8] might be explored with external non-autoregressive language models. Hidden feature distillation [11] is also a valuable extension of this paper.

6. Acknowledgment

This work is supported in part by China NSFC projects under Grants 62122050 and 62071288, and in part by Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0102. The authors would like to thank Tian tan and Wangyou Zhang for discussion.

7. References

[1] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. ICML*, 2006, pp. 369–376.

[2] E. Battenberg, J. Chen, R. Child, A. Coates, Y. G. Y. Li, H. Liu, S. Satheesh, A. Sriram, and Z. Zhu, "Exploring neural transducers for end-to-end speech recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 206–213.

[3] T. Hori, S. Watanabe, and J. R. Hershey, "Joint CTC/attention decoding for end-to-end speech recognition," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 518–529.

[4] Y. Higuchi, S. Watanabe, N. Chen, T. Ogawa, and T. Kobayashi, "Mask CTC: Non-autoregressive end-to-end ASR with CTC and mask predict," in *Interspeech 2020*. ISCA, Oct. 2020, pp. 3655–3659.

[5] Y. Higuchi, H. Inaguma, S. Watanabe, T. Ogawa, and T. Kobayashi, "Improved mask-ctc for non-autoregressive end-to-end asr," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 8363–8367.

[6] E. A. Chi, J. Salazar, and K. Kirchhoff, "Align-refine: Non-autoregressive speech recognition via iterative realignment," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, Jun. 2021, pp. 1920–1927.

[7] R. Fan, W. Chu, P. Chang, J. Xiao, and A. Alwan, "An improved single step non-autoregressive transformer for automatic speech recognition," in *Proc. Interspeech 2021*, Jun. 2021.

[8] Y. Wang, R. Liu, H. Zhang, F. Bao, and G. Gao, "Alignment-learning based single-step decoding for accurate and fast non-autoregressive speech recognition," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Feb. 2022.

[9] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Advances in Neural Information Processing Systems*, vol. 33, Oct. 2020, pp. 12 449–12 460.

[10] K. Deng, Z. Yang, S. Watanabe, Y. Higuchi, G. Cheng, and P. Zhang, "Improving non-autoregressive end-to-end speech recognition with pre-trained acoustic and language models," *arXiv:2201.10103 [cs, eess]*, Jan. 2022.

[11] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[12] M. Huang, Y. You, Z. Chen, Y. Qian, and K. Yu, "Knowledge distillation for sequence model," *Proc. Interspeech 2018*, p. 5, 2018.

[13] R. Takashima, L. Sheng, and H. Kawai, "Investigation of sequence-level knowledge distillation methods for CTC acoustic models," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 6156–6160.

[14] R. M. Munim, N. Inoue, and K. Shinoda, "Sequence-level knowledge distillation for model compression of attention-based sequence-to-sequence speech recognition," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 6151–6155.

[15] M. Ghazvininejad, O. Levy, Y. Liu, and L. Zettlemoyer, "Mask-predict: Parallel decoding of conditional masked language models," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 6112–6121.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, Dec. 2017.

[17] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented transformer for speech recognition," in *Proc. Interspeech 2020*, May 2020, pp. 5036–5040.

[18] R. Prabhavalkar, T. N. Sainath, Y. Wu, P. Nguyen, Z. Chen, C.-C. Chiu, and A. Kannan, "Minimum word error rate training for attention-based sequence-to-sequence models," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4839–4843.

[19] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "AISHELL-1: An open-source Mandarin speech corpus and a speech recognition baseline," in *2017 20th Conference of the Oriental Chapter of the International Coordinating Committee on Speech Databases and Speech I/O Systems and Assessment (O-COCOSDA)*. IEEE, 2017, pp. 1–5.

[20] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *ICASSP 2015 - 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[21] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Interspeech 2019*, pp. 2613–2617, Sep. 2019.

[22] T. Kudo and J. Richardson, "Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2018, pp. 66–71.

[23] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proc. Interspeech 2018*, 2018, pp. 2207–2211.