SELF-TRAINING FOR END-TO-END SPEECH RECOGNITION

Jacob Kahn, Ann Lee, Awni Hannun

Facebook AI Research

ABSTRACT

We revisit self-training in the context of end-to-end speech recognition. We demonstrate that training with pseudo-labels can substantially improve the accuracy of a baseline model. Key to our approach are a strong baseline acoustic and language model used to generate the pseudo-labels, filtering mechanisms tailored to common errors from sequence-to-sequence models, and a novel ensemble approach to increase pseudo-label diversity. Experiments on the LibriSpeech corpus show that with an ensemble of four models and label filtering, self-training yields a 33.9% relative improvement in WER compared with a baseline trained on 100 hours of labelled data in the noisy speech setting. In the clean speech setting, self-training recovers 59.3% of the gap between the baseline and an oracle model, which is at least 93.8% relatively higher than what previous approaches can achieve.

***Index Terms*—** speech recognition, semi-supervised, deep learning

1. INTRODUCTION

Building automatic speech recognition (ASR) systems requires a large amount of transcribed data. Compared with hybrid models, the performance of end-to-end models significantly degrades as the amount of available training data decreases [1]. Transcribing large quantities of audio is both expensive and time-consuming, and thus many semi-supervised training approaches have been proposed to take advantage of abundant unpaired audio and text data. One such approach, self-training [2], uses noisy labels generated from a model trained on a much smaller labelled data set.

We revisit self-training in the context of sequence-to-sequence models. Self-training has not been carefully studied in end-to-end speech recognition. We start from training a strong baseline acoustic model on a small paired data set and performing stable decoding [3] with a language model (LM) trained on a large-scale text corpus to generate pseudo-labels. We evaluate one heuristic and one confidence-based method for pseudo-label filtering [4–7] tailored to the mistakes often encountered with sequence-to-sequence models. In addition, we propose an ensemble approach that combines multiple models during training to improve label diversity and keep the model from being overly confident to noisy pseudo-labels.

We demonstrate the effectiveness of self-training on LibriSpeech [8], a publicly available corpus of read speech. In particular, we study the trade-off between the amount of unpaired audio data, the quality of the pseudo-labels, and the model performance. We find that in the clean speech setting, as the label quality is high, the model performance depends heavily on the amount of data. In the noisy speech setting, a proper filtering mechanism is essential for removing noisy pseudo-labels. In addition, using an ensemble of models can be complementary to filtering.

Compared with other semi-supervised methods with sequence-to-sequence models [9, 10], we show that self-training achieves a 93.8% relatively higher WER recovery rate (WRR) [11] on the clean test set, a metric indicating how much the gap between a supervised baseline and an oracle can be bridged. One goal of this work is to provide a publicly-available and reproducible benchmark to which future semi-supervised approaches in ASR can compare.

2. MODEL

Our sequence-to-sequence model is an encoder-decoder architecture with attention [12, 13]. Let $X = [X_1, \dots, X_T]$ be the frames of speech with transcription $Y = [y_1, \dots, y_U]$. The encoder maps X into a key-value hidden representation:

$$\begin{bmatrix} K \\ V \end{bmatrix} = \text{encode}(X) \tag{1}$$

where $K = [K_1, \dots, K_T]$ are the keys and $V = [V_1 \dots, V_T]$ are the values. We use a fully convolutional encoder with time-depth separable (TDS) blocks proposed in [3]. The decoder is given by

$$Q_u = RNN(y_{u-1}, Q_{u-1}) \tag{2}$$

$$S_u = \text{attend}(Q_u, K, V) \tag{3}$$

$$P(y_u \mid X, y_{<u}) = h(S_u, Q_u). \tag{4}$$

The RNN encodes the previous token and query vector Q_{u-1} to produce the next query vector. The attention mechanism produces a summary vector S_u with a simple inner product:

$$\text{attend}(K, V, Q) = V \cdot \text{softmax}\left(\frac{1}{\sqrt{d}} K^\top Q\right) \tag{5}$$

where d is the hidden dimension of K (as well as Q and V). $h(\cdot)$ computes a distribution over the output tokens.

2.1. Inference

During inference, we carry out beam search to search for the most likely hypothesis according to the sequence-to-sequence model (P_{AM}) and an external language model (P_{LM}):

$$\bar{Y} = \underset{Y}{\operatorname{argmax}} \log P_{\text{AM}}(Y \mid X) + \alpha \log P_{\text{LM}}(Y) + \beta |Y| \quad (6)$$

where α is the LM weight, and β is a token insertion term for avoiding the early stopping problem common for sequence-to-sequence models [14]. We follow the techniques in [3] to improve the efficiency and stability of the decoder. One such technique is to only propose end-of-sentence (EOS) when the corresponding probability satisfies

$$\log P_u(\text{EOS} \mid y_{<u}) > \gamma \cdot \max_{c \neq \text{EOS}} \log P_u(c \mid y_{<u}) \quad (7)$$

where γ is a hyper-parameter that can be tuned.

3. SEMI-SUPERVISED SELF-TRAINING

In a supervised learning setting, we have access to a paired data set $\mathcal{D} = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$. We train a model on $\mathcal{D}$ by maximizing the likelihood of the ground-truth transcriptions given their corresponding utterances:

$$\sum_{(X,Y) \in \mathcal{D}} \log P(Y \mid X). \quad (8)$$

In a semi-supervised setting, we have an unlabelled audio data set $\mathcal{X}$ and an unpaired text data set $\mathcal{Y}$ in addition to $\mathcal{D}$. We first train an acoustic model on $\mathcal{D}$ by maximizing the objective in Equation 8. We also train an LM on $\mathcal{Y}$. We then combine the two models to generate a pseudo-label for each unlabelled example by solving Equation 6 and obtain a pseudo paired data set $\bar{\mathcal{D}} = \{(X_i, \bar{Y}_i) \mid X_i \in \mathcal{X}\}$. A new acoustic model can be trained on the combination of $\mathcal{D}$ and $\bar{\mathcal{D}}$ with the objective

$$\sum_{(X,Y) \in \mathcal{D}} \log P(Y \mid X) + \sum_{(X,\bar{Y}) \in \bar{\mathcal{D}}} \log P(\bar{Y} \mid X). \quad (9)$$

3.1. Filtering

The pseudo-labelled data set $\bar{\mathcal{D}}$ contains noisy transcriptions. Filtering is a commonly used technique to achieve the right balance between the size of $\bar{\mathcal{D}}$ and the noise in the pseudo-labels. We design two heuristic-based filtering functions specific to sequence-to-sequence models, which can be further combined with conventional confidence-based filtering, and apply both filtering techniques on the sentence level.

Sequence-to-sequence models are known to easily fail at inference in two ways: looping and early stopping [14]. We filter for the looping by removing pseudo-labels which contain an n-gram repeated more than c times. As described in Section 2.1, we deal with early stopping by only keeping hypotheses with an EOS probability above a threshold. However, we filter examples where the beam search terminates without finding any complete hypotheses.

Additionally, for each pseudo-label, we compute the length-normalized log likelihood from the sequence-to-sequence model as the confidence score:

$$\text{ConfidenceScore}(\bar{Y}_i) = \frac{\log P_{\text{AM}}(\bar{Y}_i \mid X_i)}{|\bar{Y}_i|}$$

where $|\bar{Y}_i|$ is the number of tokens in the utterance.

3.2. Ensembles

Model combination often helps reduce word error rates in ASR. One way to utilize multiple models in self-training is to combine the model scores during inference to generate a single pseudo-labelled set with higher quality. However, as M increases, the decoding process becomes heavyweight.

Instead, we propose *sample ensemble*. Given M bootstrapped acoustic models, we generate a pseudo-labelled data set, $\bar{\mathcal{D}}_m$, for each model in parallel. We then combine all M sets of pseudo-labels with uniform weights and optimize the following objective during training

$$\sum_{(X,Y) \in \mathcal{D}} \log P(Y \mid X) + \frac{1}{M} \sum_{m=1}^{M} \sum_{(X,\bar{Y}) \in \bar{\mathcal{D}}_m} \log P(\bar{Y} \mid X).$$

In the implementation, we first train M models on $\mathcal{D}$ using different randomly initialized weights. We generate $\bar{\mathcal{D}}_m$ with hyper-parameters tuned with each model, respectively. During training, we uniformly sample a pseudo-label from one of the M models as the target in every epoch.

4. EXPERIMENTS

4.1. Data

All experiments are performed on the LibriSpeech corpus [8]. We use the "train-clean-100" set containing 100 hours of clean speech as the paired data set. We perform experiments in two settings. In the clean speech setting, we use 360 hours of clean speech in the "train-clean-360" set as the unpaired audio set, and in the noisy speech setting, we use 500 hours of noisy speech in the "train-other-500" set. We report results on the standard dev and test clean/other (noisy) sets.

The standard LM training text in LibriSpeech is derived from 14,476 public domain books [8]. To make the learning problem more realistic for self-training, we remove all books related to the acoustic training data from the LM training data, resulting in a removal of 997 books. We apply sentence segmentation using the NLTK toolkit [15] and normalize the text by lower-casing, removing punctuation except for the apostrophe, and replacing hyphens with spaces. We do not replace non-standard words with a canonical verbalized form. We find that the resulting LMs achieve comparable perplexity to LMs trained on the standard corpus on the dev sets.

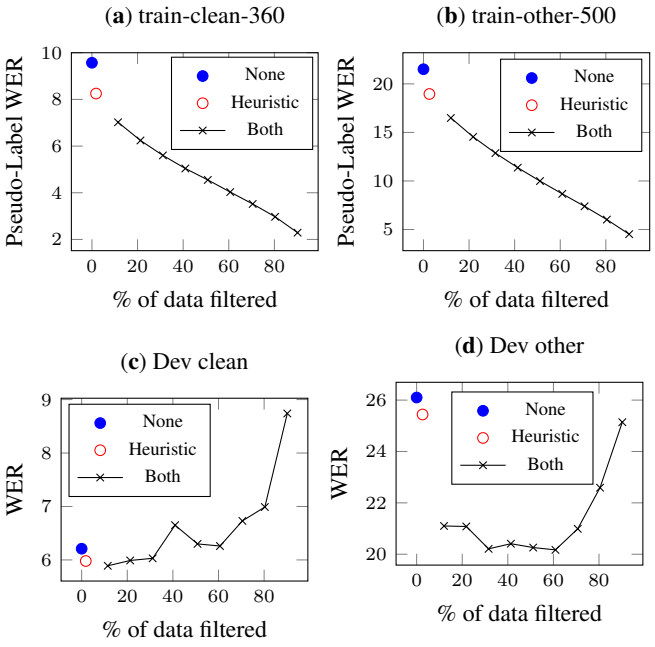

Fig. 1. Results of different filtering functions and the corresponding pseudo-label quality ((**a**), (**b**)) and model performance with LM beam search decoding ((**c**), (**d**)) in clean ((**a**), (**c**)) and noisy ((**b**), (**d**)) settings, averaged across three runs. We vary the threshold on the confidence score to filter data at various deciles. (*Both:* heuristic and confidence-based filters)

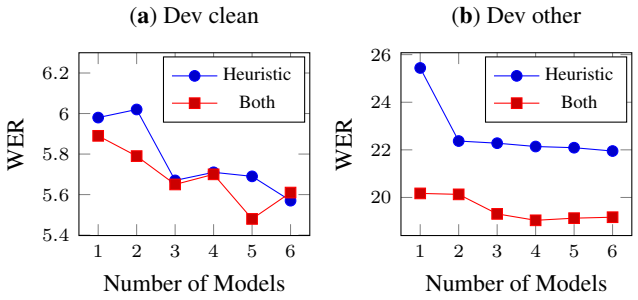

Fig. 2. WER with respect to number of models in ensemble under the clean ((**a**)) or noisy ((**b**)) setting. Results are with LM beam search decoding and averaged across three runs. (*Both:* heuristic and confidence-based filters)

4.2. Experimental Setting

Our encoder consists of nine TDS blocks in groups of three, each with 10, 14 and 16 channels and a kernel width of 21. Other architectural details are the same as [3]. We use the *SentencePiece* toolkit [16] to compute 5,000 word pieces from the transcripts in "train-clean-100" as the target tokens.

We follow the same training process as in [3] with soft-window pre-training and teacher-forcing with 20% dropout, 1% random sampling, 10% label smoothing and 1% word piece sampling for regularization. We use a single GPU with a batch size of 16 when training baselines, and 8 GPUs when training with pseudo-labels. We use SGD without momentum for 200 epochs with a learning rate of 0.05, decayed by 0.5 every 40 epochs when using one GPU or 80 epochs for 8 GPUs. Experiments are done in the *wav2letter++* framework [17].

We train a word piece convolutional LM (ConvLM) using the same model architecture and training recipe as [18]. All beam search hyper-parameters are tuned on the dev sets before generating the pseudo-labels. When training models with the combined paired and pseudo-labelled data sets, we start from random initialization instead of two-stage fine-tuning.

4.3. Results

4.3.1. Importance of Filtering

Figure 1 shows various filtering functions and the resulting amount of data, the quality of the labels and the corresponding model performance. Label quality is defined as the WER of the filtered pseudo-labels as compared to the ground truth. We apply our heuristic filtering, i.e. "no EOS + n-gram" filters, with $c = 2$ and $n = 4$ and then add confidence-based filtering on top of the filtered data set. We can see that filtering indeed improves the pseudo-label quality as we adjust the threshold on the confidence score.

In the clean setting, the heuristic filter removes 1.8% of the data, and further removal of the worst 10% of the pseudo-labels based on confidence scores results in a 5.2% relative improvement in WER on the dev clean set compared with a baseline without filtering. More aggressive filtering improves the label quality but results in worse model performance.

In the noisy setting, removing the worst 10% of the pseudo-labels results in a significant reduction in WER, and the best performance comes from filtering 60% of the labels with a WER 22.7% relative lower on the dev other set compared with no filtering. Filtering more data leads to the same degradation in model performance as in the clean setting.

4.3.2. Model Ensembles

Figure 2 shows WER as a function of the number of models in the ensemble on the dev sets for both clean and noisy settings. We can see that combining multiple models improves the performance, especially for the noisy setting, where we obtain a 13.7% relative improvement with six models and heuristic filtering. One possible explanation is that since the *sample ensemble* uses different transcripts for the same utterance at training time, this keeps the model from being overly confident in a noisy pseudo-label. We also show that the two filtering techniques can be combined with ensembles effectively. In the noisy setting, model ensembles with both filterings improve WER by 27.0% relative compared with a single model without any filtering (Figure 1(d)).

4.3.3. Comparison with Literature

Table 1 summarizes our best results, as well as the supervised baseline and the oracle models trained with ground-truth tran-

Method	No LM				With LM			
	Dev WER		Test WER (WRR)		Dev WER		Test WER (WRR)	
	clean	other	clean	other	clean	other	clean	other
Baseline Paired 100hr	14.00	37.02	14.85	39.95	7.78	28.15	8.06	30.44
Paired 100hr + Unpaired 360hr clean speech								
Oracle	7.20	25.32	7.99	26.59	3.98	17.00	4.23	17.36
Single Pseudo	9.61	29.72	10.27 (66.8%)	30.50 (70.7%)	5.84	21.86	6.46 (41.8%)	22.90 (57.6%)
Ensemble (5 models)	9.00	27.74	**9.62** (76.2%)	29.53 (78.0%)	5.41	20.31	**5.79** (59.3%)	21.63 (67.4%)
Paired 100hr + Unpaired 500hr noisy speech								
Oracle	6.90	17.55	7.09	18.36	3.74	10.49	3.83	11.28
Single Pseudo	10.90	28.37	11.48 (43.4%)	29.73 (47.3%)	6.38	19.98	6.56 (35.5%)	22.09 (43.6%)
Ensemble (4 models)	10.41	27.00	10.50 (56.1%)	**29.25** (49.6%)	6.01	18.95	6.20 (44.0%)	**20.11** (53.9%)

Table 1. Best results from single runs tuned on the dev sets. The best filtering setup found in Section 4.3.1 is applied.

Method	Text (# words)	No LM Test clean WER (WRR)	With LM Test clean WER (WRR)
Cycle TTE [9]	4.8M	21.5 (27.6%)	19.5 (30.6%*)
ASR+TTS [10]	3.6M	17.5 (38.0%)	16.6 (-)
this work	842.5M	**9.62** (**76.2%**)	**5.79** (**59.3%**)

Table 2. A comparison with previous work using 100hr paired data and 360hr unpaired audio. WRR is computed with the baseline and oracle WER from the original work if available. (*: The oracle WER is without LM decoding, so the WRR is an upper bound estimation.)

scriptions. We present results from both AM only greedy decoding and LM beam search decoding to demonstrate the full potential of self-training. In addition to WER, we report WER recovery rate (WRR) [11] to demonstrate how much gap between the baseline and the oracle that we can bridge with pseudo-labels. WRR is defined as

$$\frac{\text{baseline WER} - \text{semi-supervised WER}}{\text{baseline WER} - \text{oracle WER}}.$$

When decoded with an external LM, our best model achieves a WRR over 50% in both clean and noisy speech settings.

Table 2 compares our approach with other semi-supervised learning methods with sequence-to-sequence models that use the same audio data setup. We see that our conventional pseudo-labelling approach together with filtering and ensemble produces a WER at least 65.1% relatively lower than the previously best results. The gain comes from the strong baseline model with TDS-based encoders [3] to generate the pseudo-labels, and a much larger unpaired text corpus, which we believe is easy to obtain in a real-world setting. As a comparison, the baseline WER on the test clean set is above 20 in [9, 10]. However, even with a strong baseline, we achieve a WRR at least 93.8% relatively higher than other methods.

5. RELATED WORK

In speech recognition, self-training has been explored in hybrid systems [4–7, 19]. Prior work mainly focuses on different ways of data filtering to improve pseudo-label quality, e.g. confidence-based filtering [4, 5] and agreement-based selection [20], which also takes advantage of multiple systems. The data selection process can take place at different levels ranging from frames to utterances [6, 7]. In [21], the output probability of a teacher model is used as soft pseudo-labels to train a student model. Training with pseudo-labels can give an improvement to WER not only for low-resource languages [6, 7] but also on large-scale data sets [21].

Recently-proposed semi-supervised approaches for end-to-end speech recognition take advantage of text-to-speech (TTS) modules to generate synthetic data from unpaired text [22] or introduce a cycle-consistency loss between the input and the output of an ASR+TTS pipeline [9, 10]. Alternatively, inter-domain loss is proposed to constrain speech and text in the same embedding space [23]. In this work, we demonstrate that the self-training approach is simple yet effective with end-to-end systems.

6. CONCLUSION

We have shown that self-training can yield substantial improvements for end-to-end systems over a strong baseline model by leveraging a large unlabelled data set. We show that filtering mechanisms tailored to the types of mistakes encountered with sequence-to-sequence models as well as an ensemble of models can further improve the accuracy gains from self-training. Our experiments on LibriSpeech have set forth a strong baseline model and a reproducible semi-supervised learning setting for which new and more sophisticated approaches can be evaluated.

7. ACKNOWLEDGEMENTS

Thanks to Tatiana Likhomanenko, Qiantong Xu, Ronan Collobert and Gabriel Synnaeve for their help with this work.

8. REFERENCES

[1] Christoph Lüscher, Eugen Beck, Kazuki Irie, Markus Kitza, Wilfried Michel, Albert Zeyer, Ralf Schlüter, and Hermann Ney, "RWTH ASR systems for LibriSpeech: Hybrid vs attention," in *Interspeech*, 2019.

[2] H Scudder, "Probability of error of some adaptive pattern-recognition machines," *IEEE Transactions on Information Theory*, vol. 11, no. 3, pp. 363–371, 1965.

[3] Awni Hannun, Ann Lee, Qiantong Xu, and Ronan Collobert, "Sequence-to-sequence speech recognition with time-depth separable convolutions," in *Interspeech*, 2019.

[4] Delphine Charlet, "Confidence-measure-driven unsupervised incremental adaptation for hmm-based speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2001.

[5] Frank Wessel and Hermann Ney, "Unsupervised training of acoustic models for large vocabulary continuous speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 1, pp. 23–31, 2004.

[6] Karel Veselỳ, Mirko Hannemann, and Lukáš Burget, "Semi-supervised training of deep neural networks," in *Workshop on Automatic Speech Recognition and Understanding*, 2013.

[7] Karel Veselỳ, Lukás Burget, and Jan Cernockỳ, "Semi-supervised DNN training with word selection for ASR," in *Interspeech*, 2017.

[8] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "LibriSpeech: an ASR corpus based on public domain audio books," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

[9] Takaaki Hori, Ramon Astudillo, Tomoki Hayashi, Yu Zhang, Shinji Watanabe, and Jonathan Le Roux, "Cycle-consistency training for end-to-end speech recognition," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[10] Murali Karthick Baskar, Shinji Watanabe, Ramon Astudillo, Takaaki Hori, Lukáš Burget, and Jan Černockỳ, "Semi-supervised sequence-to-sequence ASR using unpaired speech and text," in *Interspeech*, 2019.

[11] Jeff Ma and Richard Schwartz, "Unsupervised versus supervised training of acoustic models," in *Interspeech*, 2008.

[12] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," in *International Conference on Learning Representations*, 2015.

[13] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," in *EMNLP*, 2014.

[14] Jan Chorowski and Navdeep Jaitly, "Towards better decoding and language model integration in sequence to sequence models," in *Interspeech*, 2017.

[15] Edward Loper and Steven Bird, "NLTK: The natural language toolkit," in *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, 2002, pp. 63–70.

[16] Taku Kudo and John Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *EMNLP: System Demonstrations*, 2018.

[17] Vineel Pratap, Awni Hannun, Qiantong Xu, Jeff Cai, Jacob Kahn, Gabriel Synnaeve, Vitaliy Liptchinsky, and Ronan Collobert, "Wav2letter++: A fast open-source speech recognition system," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[18] Neil Zeghidour, Qiantong Xu, Vitaliy Liptchinsky, Nicolas Usunier, Gabriel Synnaeve, and Ronan Collobert, "Fully convolutional speech recognition," *arXiv preprint arXiv:1812.06864*, 2018.

[19] Thomas Kemp and Alex Waibel, "Unsupervised training of a speech recognizer: Recent experiments," in *EUROSPEECH*, 1999.

[20] Félix de Chaumont Quitry, Asa Oines, Pedro Moreno, and Eugene Weinstein, "High quality agreement-based semi-supervised training data for acoustic modeling," in *Spoken Language Technology Workshop (SLT)*, 2016.

[21] Sree Hari Krishnan Parthasarathi and Nikko Strom, "Lessons from building acoustic models with a million hours of speech," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019.

[22] Tomoki Hayashi, Shinji Watanabe, Yu Zhang, Tomoki Toda, Takaaki Hori, Ramon Astudillo, and Kazuya Takeda, "Back-translation-style data augmentation for end-to-end ASR," in *Spoken Language Technology Workshop (SLT)*, 2018.

[23] Shigeki Karita, Shinji Watanabe, Tomoharu Iwata, Atsunori Ogawa, and Marc Delcroix, "Semi-supervised end-to-end speech recognition," in *Interspeech*, 2018.