EFFICIENT SEQUENCE TRAINING OF ATTENTION MODELS USING APPROXIMATIVE RECOMBINATION

Nils-Philipp Wynands[1], Wilfried Michel[1,2], Jan Rosendahl[1], Ralf Schlüter[1,2], Hermann Ney[1,2]

[1]Human Language Technology and Pattern Recognition, Computer Science Department,
RWTH Aachen University, 52062 Aachen, Germany,
[2]AppTek GmbH, 52062 Aachen, Germany

nils-philipp.wynands@rwth-aachen.de, {michel,rosendahl,schlueter,ney}@cs.rwth-aachen.de

ABSTRACT

Sequence discriminative training is a great tool to improve the performance of an automatic speech recognition system. It does, however, necessitate a sum over all possible word sequences, which is intractable to compute in practice. Current state-of-the-art systems with unlimited label context circumvent this problem by limiting the summation to an n-best list of relevant competing hypotheses obtained from beam search.

This work proposes to perform (approximative) recombinations of hypotheses during beam search, if they share a common local history. The error that is incurred by the approximation is analyzed and it is shown that using this technique the effective beam size can be increased by several orders of magnitude without significantly increasing the computational requirements. Lastly, it is shown that this technique can be used to effectively perform sequence discriminative training for attention-based encoder-decoder acoustic models on the LibriSpeech task.

Index Terms— beam search, global normalization, language model integration, lattice, sequence training

1. INTRODUCTION

State-of-the-art approaches to *automatic speech recognition* (ASR) make use of *acoustic models* (AM) whose parameters are estimated from parallel data consisting of audio signals and corresponding transcriptions. The outcome of the model training depends on the available amount of such parallel data. But the amount of available parallel data is usually quite limited, compared to the publicly available amount of unimodal data (i.e. audio or textual). Consequently, there are attempts to make use of unimodal data to improve the performance of AMs by unsupervised pretraining [1] or combining AMs with external *language models* (LM) which have been trained on text only data [2].

Previous work [3] shows that integrating LMs into the training of attention-based encoder-decoder models via a log-linear combination followed by a renormalization on sequence-level yields promising improvements similar to the maximum mutual information (MMI) training of hybrid models. However, since the sum over all possible word sequences used in the renormalization cannot be entirely computed in practice, it is usually approximated by a sum over an n-best list obtained from *beam search* (BS) with finite beam size. The number of sequences which can be aggregated through this approach is usually limited to 2-8 competing hypotheses for both attention [4] and transducer models [5, 6].

In the hybrid approach, this limitation is alleviated by the use of lattices [7], where hypotheses sharing a common history determined by the LM context length are recombined. The lattice approach for recognition has been explored in [8] for temporal convolution models, but it is not directly applicable for general *attention-based encoder-decoder* (AED) models.

In this work an extension to the n-best list approach is proposed. Introducing recombination operations in the standard beam search procedure the n-best list is transformed into a lattice which contains a denser and much richer representation of the search space. Using lattices, the number of sequences considered for the approximation of the renormalization can be increased significantly.

2. TRAINING CRITERION

In the following, let $w_1^N := w_1...w_n...w_N$ denote a word sequence of length $N \in \mathbb{N}$, and let $x_1^T := x_1...x_t...x_T$ denote an input signal sequence of length $T \in \mathbb{N}$.

In this work, an attention-based AM $p_{\mathrm{AM}}(w_1^N|x_1^T)$ and an external LM $p_{\mathrm{LM}}(w_1^N)$ are combined via log-linear model combination. The (normalized) probability of the *combined model* is given by

$$p_c(w_1^N|x_1^T) = \frac{p_{\mathrm{AM}}^{\alpha}(w_1^N|x_1^T) \cdot p_{\mathrm{LM}}^{\beta}(w_1^N)}{\sum_{\tilde{N},\tilde{w}_1^{\tilde{N}}} p_{\mathrm{AM}}^{\alpha}(\tilde{w}_1^{\tilde{N}}|x_1^T) \cdot p_{\mathrm{LM}}^{\beta}(\tilde{w}_1^{\tilde{N}})} \qquad (1)$$

where the influence of AM and LM on the output of the combined model is scaled by the AM scale $\alpha \in \mathbb{R}^+$ and the LM scale $\beta \in \mathbb{R}^+$. The sum in the denominator contains all possible word sequences and is infeasible to compute in practice.

For recognition, the normalization term is not needed, as it does not influence the result. In training, however, the sum does appear in the training criterion

$$F = \log p_c(w_1^N | x_1^T). \tag{2}$$

2.1. Approximative Lattice Recombination

In the hybrid approach, sequence training can be performed efficiently with the help of lattices. Here, the acoustic and language model only take a limited number k of past words into account. So if the last k words of two hypotheses match they can be recombined exactly.

For attention models or transducers with unlimited history context, this can never happen. It is possible to force a recombination with different label histories w_1^n and $\tilde{w}_1^n$ under the approximation that for every following step m

$$p_c(w | w_1^m, x_1^T) \approx p_c(w | w_{n+1}^m, \tilde{w}_1^n, x_1^T) \tag{3}$$

holds. In the following the proposed procedure to exploit this approximation during beam search is described.

Let the *recombination history limit* $k \in \mathbb{N}$ be the minimum number of tokens of common history after which the approximation of Equation 3 is assumed to be sufficiently well met. Then let $B_n(w_{n-k}^{n-1})$ denote the set of sequences within the beam at decoding step n, which share the common history w_{n-k}^{n-1}.

Before the hypotheses are expanded with the predictions for step n, candidates for recombination are identified. For each set of hypotheses that can be recombined, the sequence with the highest score so far is selected

$$\hat{w}_1^{n-1} = \underset{w_1^{n-1} \in B_n(w_{n-k}^{n-1})}{\arg\max} \; p(w_1^{n-1} | x_1^T) \tag{4}$$

and the score mass of all other hypotheses in $B_n(w_{n-k}^{n-1})$ is combined into the hypothesis of $\hat{w}_1^{n-1}$. Then the other sequences are removed from the search space and the hypothesis of $\hat{w}_1^{n-1}$ now holds the probability mass

$$\hat{p}(\hat{w}_1^{n-1} | x_1^T) = \sum_{w_1^{n-1} \in B_n(w_{n-k}^{n-1})} p(w_1^{n-1} | x_1^T). \tag{5}$$

Subsequently, the branch expansion continues with the remaining hypotheses and altered scores. Once all sequences in the beam are terminated with a sentence-end token, the search terminates and the probability mass is summed to approximate the sum of the scores of all possible word sequences.

An exemplary lattice built by this search procedure together with the sequences it contains is illustrated in Figure 1.

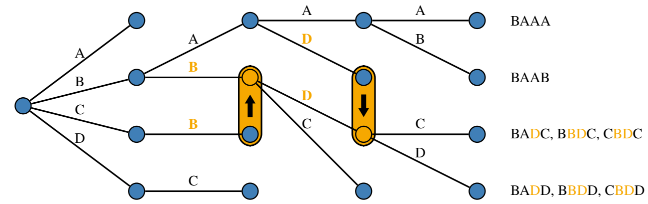

Fig. 1: Illustration of lattice recombination with beam size $b = 4$ and recombination history limit $k = 1$. The search progresses from left to right. The matching histories are highlighted in orange. The sequences considered by the search are listed on the right.

3. EXPERIMENTAL SETUP

All experiments use the Sisyphus [9] workflow manager and RETURNN [10]. The configuration files of the experiments are available online. [1]

The experiments of this work are conducted on the Librispeech task and use two kinds of models, one attention-based encoder-decoder model with a decoder with limited label context length and one with unlimited label context length. In both cases, the same encoder architecture is used for the AM, which consists of 2 CNN layers, followed by 6 BLSTM layers which perform time sub-sampling by a factor of 6 via max-pooling. As input, the encoder receives 40 dimensional MFCC features which are perturbed by a variant of SpecAugment [11]. Further, in both cases, MLP-style attention with weight feedback is used. Note that the weight feedback adds some recurrence to the decoder of the AM with limited context length. The output vocabulary consists of 10k grapheme BPE tokens.

For the AMs with unlimited context length, the decoder is based on a single LSTM layer with 1000 units [3, 12, 13]. For the AMs with limited context length, the decoder consists of a single FFNN layer, which receives an attention vector as well as the last 5 target embeddings as input [13].

For the construction of combined models, the LSTM-decoder based AM is combined with an LSTM LM consisting of a single LSTM layer and the FFNN AM is combined with a FFNN LM which consists of 3 FFNN layers and also has a label context length of 5. Both LMs were trained on the audio transcriptions only. While an LSTM LM or an FFNN LM respectively is used for training, a Transformer LM consisting of 24 layers and trained on additional text data is used during decoding in any case. The LMs operate on the same vocabulary as the acoustic models do.

AMs and LMs are first trained separately with the *cross-entropy* (CE) criterion until they converge, before they are used to initialize the combined models. The AM training includes a pre-training scheme with gradually increasing layer sizes and an additional CTC-loss on the encoder outputs [12].

[1] `https://github.com/rwth-i6/returnn-experiments/tree/master/2022-approx-recombination`

Table 1: Comparison of lattices statistics obtained with different recombination history limits k by averaging searches over the first 1024 segments of the LibriSpeech dev-other set. The rows where $k = \infty$ correspond to n-best lists.

(a) Combined model with unlimited context length.

history limit k	log-score mass	number of sequences	number of recomb.	Euclid. dist.
1	-7.098	$1.2 \cdot 10^{13}$	15.8	0.195
2	-8.032	$1.2 \cdot 10^{7}$	10.0	0.106
3	-8.522	$3.1 \cdot 10^{3}$	7.5	0.077
4	-8.777	$1.9 \cdot 10^{3}$	6.0	0.060
6	-9.166	$4.3 \cdot 10^{1}$	4.1	0.041
8	-9.299	$2.3 \cdot 10^{1}$	3.1	0.028
10	-9.466	$1.7 \cdot 10^{1}$	1.0	0.023
12	-9.560	$1.4 \cdot 10^{1}$	1.9	0.018
∞	-9.907	$0.8 \cdot 10^{1}$	0.0	–

(b) Combined model with limited context length of 5.

history limit k	log-score mass	number of sequences	number of recomb.	Euclid. dist.
1	-6.836	$7.2 \cdot 10^{12}$	15.9	0.2392
2	-7.824	$3.5 \cdot 10^{7}$	9.9	0.1363
4	-8.639	$2.8 \cdot 10^{3}$	5.9	0.0842
5	-8.832	$6.3 \cdot 10^{2}$	4.8	0.0028
8	-9.270	$2.7 \cdot 10^{1}$	3.0	0.0002
∞	-9.879	$0.8 \cdot 10^{1}$	0.0	–

The combined model is then fine tuned with the sequence discriminative objective function in Equation 2 using different recombination history limits k. During the training of the combined models, only the parameters of the underlying AMs are updated.

In training, an AM scale of $\alpha = 0.1$ and an LM scale of $\beta = 0.035$ is used, which were found to be optimal in [3]. For decoding the scales are individually tuned on the respective dev sets.

4. RESULTS

4.1. Lattice Analysis

To investigate how the lattice approximation behaves in practice compared to the standard n-best list, lattice statistics are extracted for the combined model both with unlimited and with limited context length of 5. In both cases, the combined model just after initialization with the respective underlying AM and LM is used.

For the experiment, beam search is conducted over each of the first 1024 segments of the LibriSpeech dev-other set for different recombination history limits k. Additionally, a beam search without recombination is conducted for comparison, which in the following is reported as $k = \infty$. The beam size for each search is set to $b = 8$.

For each search the number of word sequences that can be obtained from the final lattice, the amount of aggregated score mass, and the number of recombinations are measured. On each recombination, the context with the highest sequence score so far is selected to replace the contexts of the other recombined branches. In future steps the correct probability distribution of these branches is displaced by an approximative distribution with the updated context. The difference between these two distributions is estimated by averaging the Euclidean distance between the best-context distribution and each recombined distribution.

$$d = \sum_{w} \left| p_n(w|\hat{w}_1^{n-1}, x_1^T) - p_n(w|w_1^{n-1}, x_1^T) \right|^2 \qquad (6)$$

The statistics obtained from each segment search are averaged for each k and displayed in Table 1.

As to be expected, lowering k increases the number of recombinations which are made during search, which in turn increases the number of sequences considered and the score mass aggregated through a search. In case of single word recombination history limit compared to the case without recombination ($k = \infty$), the average number of considered sequences is increased by a factor of 10^{12} and a 20-fold greater average score mass can be aggregated. However, lowering k also worsens the approximations, as seen by the increasing Euclidean distances.

For the limited context model, the Euclidean distance lies above the the distance measured for the unlimited context model for $k < 5$. As the recombination history limit reaches the model context length, a significant drop in the distance is observed and it is now more than one order of magnitude lower than the comparable distance of the full context model. The distance does, however, not immediately reach exactly zero. This is due to the attention feedback of the underlying AM which applies some recurrence and hence some dependence on the complete context to the model.

4.2. Training results

In the next step, lattice recombination is used to approximate the denominator in Equation 1 during training. The trainings are conducted using the beam sizes $b = 8$ and $b = 4$. All tables also include the results of using n-best lists for the approximation of the denominator sum in the row $k = \infty$ for comparison.

Unlimited Context Models For the model with unlimited context length, multiple recombination history limits k are evaluated. The training results are listed in Table 2.

It can be seen that in the limit of large k, the result of n-best lists can be recovered. Reducing k degrades the *word*

Table 2: Comparing performances of combined models with unlimited context length trained with different recombination history limits k, where $k = \infty$ means using n-best list.

(a) Denom. beam size $b = 8$

history limit k	dev WER[%] clean	other
1	2.2	6.2
2	2.1	6.2
3	2.1	6.1
4	2.1	6.0
6	2.1	5.9
8	2.1	5.9
10	2.1	5.8
12	2.1	5.8
∞	2.1	5.8

(b) Denom. beam size $b = 4$

history limit k	dev WER[%] clean	other
1	2.2	6.2
2	2.1	6.0
3	2.1	6.0
4	2.1	5.9
6	2.1	6.0
8	2.1	6.0
10	2.1	5.9
12	2.1	6.0
∞	2.1	6.0

Table 3: Comparing performance of combined models with limited context length of 5 trained with recombination history limit of $k = 5$ and $k = \infty$, which corresponds to using n-best list.

(a) Denom. beam size $b = 8$

history limit k	dev WER[%] clean	other
5	2.3	6.2
∞	2.3	6.4

(b) Denom. beam size $b = 4$

history limit k	dev WER[%] clean	other
5	2.3	6.2
∞	2.4	6.3

error rate (WER) of the final model on the LibriSpeech dev sets. The benefit of a larger effective beam size is surpassed by the error that is incurred by the stronger approximations.

In case of the models trained with a beam size of $b = 4$, a small degradation compared to $b = 8$ is observed. Here, reducing the recombination history limit does not change the results significantly. Even with a higher effective beam size, the $b = 8$ result cannot be recovered.

Limited Context Models In case of the combined model with limited context length, the recombination history limit is chosen to be equal to the model context length $k = 5$. The results are listed in Table 3.

Using lattices with a recombination history limit of $k = 5$ yields small improvements over the use of n-best lists on dev-other and similar results on dev-clean. With the smaller beam size $b = 4$ the same result as with $b = 8$ is achieved but also no clear degradation is visible in the case of n-best lists.

The results from the initial model without sequence training together with the test results of the combined models which performed best on the dev sets are listed in Table 4 and are consistent with the observations made on the dev sets.

Table 4: Comparing the performances of the discriminatively trained combined models which perform best on the dev sets. WER of the model used for initialization (init) is included for reference.

model context	history limit k	dev WER[%] clean	other	test WER[%] clean	other
limited	init	2.6	6.7	3.0	7.1
	5	2.3	6.2	2.6	6.6
	∞	2.3	6.4	2.6	6.7
unlimited	init	2.4	6.7	2.7	7.0
	12	2.1	5.8	2.3	6.5
	∞	2.1	5.8	2.3	6.5

5. CONCLUSION

In this paper, *approximative recombination* is used to improve the summation of the scores of all competing word sequences, which is used in sequence discriminative training of attention-based encoder-decoder models.

Using lattices, the average number of sequences considered for the summation can be increased by a factor of up to 10^{12} and an up to 20-fold larger average score mass can be aggregated compared to standard n-best lists (cf. Table 1).

Sequence training with both, n-best lists and lattices, show good improvement over the cross entropy baseline model. In the limit of large recombination history limits k, the n-best list result can be recovered, but for smaller k only the limited context FFNN decoder model shows additional improvement over standard beam search.

Nevertheless, approximative recombination may be a useful tool for applications that benefit from the larger search space coverage without being handicapped by the additional approximation, such as lattice rescoring or keyword spotting.

6. ACKNOWLEDGMENTS

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 694537, project "SEQCLAS"). The work reflects only the authors' views and none of the funding parties is responsible for any use that may be made of the information it contains.

The authors would like to thank Mohammad Zeineldeen and Aleksandr Glushko for providing the baseline AMs and helpful discussions.

7. REFERENCES

[1] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli, "wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems (NeurIPS)*, December 2020.

[2] Çaglar Gülçehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loïc Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "On Using Monolingual Corpora in Neural Machine Translation," in *Computer Speech & Language*, September 2017, vol. 45, pp. 137–148.

[3] Wilfried Michel, Ralf Schlüter, and Hermann Ney, "Early Stage LM Integration Using Local and Global Log-Linear Combination," in *Proc. Interspeech*, October 2020, p. 3605–3609.

[4] Rohit Prabhavalkar, Tara N. Sainath, Yonghui Wu, Patrick Nguyen, Zhifeng Chen, Chung-Cheng Chiu, and Anjuli Kannan, "Minimum Word Error Rate Training for Attention-Based Sequence-to-Sequence Models," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 4839–4843.

[5] Chao Weng, Chengzhu Yu, Jia Cui, Chunlei Zhang, and Dong Yu, "Minimum Bayes Risk Training of RNN-Transducer for End-to-End Speech Recognition," in *Proc. Interspeech*, October 2020, pp. 966–970.

[6] Liang Lu, Zhong Meng, Naoyuki Kanda, Jinyu Li, and Yifan Gong, "On Minimum Word Error Rate Training of the Hybrid Autoregressive Transducer," in *Proc. Interspeech 2021*, August 2021, pp. 3435–3439.

[7] Brian Kingsbury, "Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*. April 2009, pp. 3761–3764, IEEE.

[8] Michal Zapotoczny, Piotr Pietrzak, Adrian Lancucki, and Jan Chorowski, "Lattice Generation in Attention-Based Speech Recognition Models," in *Proc. Interspeech*. September 2019, pp. 2225–2229, ISCA.

[9] Jan-Thorsten Peter, Eugen Beck, and Hermann Ney, "Sisyphus, a workflow manager designed for machine translation and automatic speech recognition," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Brussels, Belgium, Nov. 2018, pp. 84–89, Association for Computational Linguistics.

[10] Patrick Doetsch, Albert Zeyer, Paul Voigtlaender, Ilia Kulikov, Ralf Schlüter, and Hermann Ney, "RETURNN: the RWTH extensible training framework for universal recurrent neural networks," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 5345–5349.

[11] Daniel S. Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D. Cubuk, and Quoc V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," *Proc. Interspeech 2019*, pp. 2613–2617, Sep 2019.

[12] Albert Zeyer, Parnia Bahar, Kazuki Irie, R. Schlüter, and H. Ney, "A Comparison of Transformer and LSTM Encoder Decoder Models for ASR," *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 8–15, December 2019.

[13] Mohammad Zeineldeen, Aleksandr Glushko, Wilfried Michel, Albert Zeyer, Ralf Schlüter, and Hermann Ney, "Investigating Methods to Improve Language Model Integration for Attention-based Encoder-Decoder ASR Models," in *Proc. Interspeech*, August 2021, pp. 2856–2860.