Equivalence of Segmental and Neural Transducer Modeling: A Proof of Concept

Wei Zhou[1,2], Albert Zeyer[1,2], André Merboldt[1], Ralf Schlüter[1,2], Hermann Ney[1,2]

[1]Human Language Technology and Pattern Recognition, Computer Science Department, RWTH Aachen University, 52074 Aachen, Germany
[2]AppTek GmbH, 52062 Aachen, Germany

{zhou, zeyer, schlueter, ney}@cs.rwth-aachen.de, andre.merboldt@rwth-aachen.de

Abstract

With the advent of direct models in automatic speech recognition (ASR), the formerly prevalent frame-wise acoustic modeling based on hidden Markov models (HMM) diversified into a number of modeling architectures like encoder-decoder attention models, transducer models and segmental models (direct HMM). While transducer models stay with a frame-level model definition, segmental models are defined on the level of label segments directly. While (soft-)attention-based models avoid explicit alignment, transducer and segmental approach internally do model alignment, either by segment hypotheses or, more implicitly, by emitting so-called blank symbols. In this work, we prove that the widely used class of RNN-Transducer models and segmental models (direct HMM) are equivalent and therefore show equal modeling power. It is shown that blank probabilities translate into segment length probabilities and vice versa. In addition, we provide initial experiments investigating decoding and beam-pruning, comparing time-synchronous and label-/segment-synchronous search strategies and their properties using the same underlying model.

Index Terms: automatic speech recognition, transducer, RNN-T, segmental model, direct HMM

1. Introduction

Most acoustic models in speech recognition can be categorized into being defined on the level of each time frame or directly on the level of labels and/or segments. The hybrid neural network (NN) - hidden Markov model (HMM) [1, 2] and extensions [3], CTC [4] and their generalized transducer variants [5–10] are all defined on the time-frame level. The attention-based encoder-decoder model [11–15] and in general segmental models [16–20] are defined on a label and/or segment level. Segmental models generalize from the encoder-decoder model by introducing an explicit latent variable which usually represents the temporal position of a label [20]. Such explicit representation of the time is needed to enable monotonicity and to potentially allow for online streaming. While [8] briefly mentioned the duration interpretation of the blank probability of a transducer model, the general interrelation between time-synchronous models and segmental models are not well studied in the literature.

In this work, we prove the equivalence of transducer models and segmental models, with the latter introducing an explicit representation for the temporal position or boundaries per label segment as a latent variable. This equivalence implies that we can use both label-synchronous or time-synchronous beam search decoding for either case. We provide initial experiments comparing label-synchronous and time-synchronous beam search decoding for the transducer models introduced in [9, 10].

2. Model Equivalence

Let $x_1^{T'}$ denote the input feature sequence and $h_1^T = f^{\text{enc}}(x_1^{T'})$ denote the encoder output, which transforms the input into a sequence of high-level representations. In general, $T \le T'$ due to optional sub-sampling in the encoder. Let $a_1^S \in V^S$ denote a label sequence of length S from a vocabulary V. The general sequence-to-sequence models target at the following sequence posterior probability:

$$p(a_1^S \mid x_1^{T'}) = p(a_1^S \mid h_1^T) \tag{1}$$

In the following, we show the equivalence between segmental modeling and transducer modeling of Eq. (1) under the constraint that all T encoder output frames have to be consumed (denoted as constraint-T). Without loss of generality, we show that a transducer model based on the RNN-T topology [5] can be rewritten into a segmental model which allows zero-frame segments, and also that a segmental model can be rewritten into an RNN-T model. For both rewriting directions, we provide the corresponding transformation equations. Also the case of assuming transducer models with strict monotonicity as in [6, 7], is covered as a special case of the equivalence, which leads to a segmental model with minimum segment length of one frame.

2.1. Segmental Model

In a segmental model, the label sequence posterior from Eq. (1) can be formulated as:

$$p(a_1^S \mid h_1^T) = \sum_{t_1^S} p(a_1^{S+1}, t_1^{S+1} \mid h_1^T) \tag{2}$$

$$= \sum_{t_1^S} \prod_{s=1}^{S+1} p(a_s, t_s \mid a_1^{s-1}, t_1^{s-1}, h_1^T)$$

$$= \sum_{t_1^S} \prod_{s=1}^{S+1} p(t_s \mid a_1^{s-1}, t_1^{s-1}, h_1^T) \cdot p(a_s \mid a_1^{s-1}, t_1^s, h_1^T)$$

where t_s are the segment boundaries of label a_s under the monotonicity constraints $t_{s-1} \le t_s \le T$ for $1 \le s \le S + 1$. Here we explicitly introduce a final sentence end label $a_{S+1} = \#$, and define $t_0 \equiv 1$ and $t_{S+1} \equiv T$ for the constraint-T.

2.2. Transducer Model

Within a transducer modeling approach using the RNN-T topology, Eq. (1) can be formulated as:

$$p(a_1^S \mid h_1^T) = \sum_{y_1^U : a_1^S} p(y_1^U \mid h_1^T)$$
$$= \sum_{y_1^U : a_1^S} \prod_{u=1}^{U=T+S} q(y_u \mid y_1^{u-1}, h_1^T) \qquad (3)$$

where y_1^U is the blank ϵ-augmented alignment sequence of a_1^S and q is a probability distribution defined over $\bar{V} = V \cup \{\epsilon\}$. The RNN-T label topology is composed of horizontal transitions for $y_u = \epsilon$ and vertical transitions for $y_u \neq \epsilon$, where a final blank transition at step U is always presented for termination.

2.3. From Transducer to Segmental Model

We postulate that any transducer model using the RNN-T topology as defined in Eq. (3) can be rewritten into an equivalent segmental model that defines exactly the same overall label posterior distribution. The following transformation equations show the resulting segmental model written in terms of the transducer model:

$$p(t_s \mid a_1^{s-1}, t_1^{s-1}, h_1^T) = \prod_{t=t_{s-1}}^{t_s-1} q(y_{t+s-1} = \epsilon \mid y_1^{t+s-2}, h_1^T)$$
$$\cdot (1 - q(y_{t_s+s-1} = \epsilon \mid y_1^{t_s+s-2}, h_1^T)) \quad (4)$$

and

$$p(a_s \mid a_1^{s-1}, t_1^s, h_1^T) = \frac{q(y_{t_s+s-1} = a_s \mid y_1^{t_s+s-2}, h_1^T)}{1 - q(y_{t_s+s-1} = \epsilon \mid y_1^{t_s+s-2}, h_1^T)} \qquad (5)$$

with the alignment sequence y_1^U defined as follows:

$$y_u = \begin{cases} a_s & \text{iff } \exists s : u = t_s + s - 1 \\ \epsilon & \text{otherwise} \end{cases} \quad \forall u = 1, \dots, T+S.$$

Separately, we compute the probability for sentence end in the segmental model as follows:

$$p(a_{S+1} = \# \mid a_1^S, t_1^{S+1}, h_1^T) = \frac{q(y_{T+S} = \epsilon \mid y_1^{T+S-1}, h_1^T)}{1 - q(y_{T+S} = \epsilon \mid y_1^{T+S-1}, h_1^T)}. \qquad (6)$$

Similar as in [21], the segment duration probability in Eq. (4) is represented as a Bernoulli-like length distribution by regarding ϵ as a pooled state for segment continuation. Substituting Eqs. (4) and (5) into the segmental model defined in Eq. (2) leads to the transducer model defined in Eq. (3). This shows that using Eqs. (4) and (5), any transducer model can be rewritten into a segmental model providing an identical label posterior distribution.

2.4. From Segmental Model to Transducer

Also, we postulate that any segmental model as defined in Sec. 2.1 can be rewritten into an equivalent transducer model using the RNN-T topology that also defines exactly the same overall label posterior distribution. The following transformation equations show the resulting transducer model written in terms of the segmental model:

$$q(y_{u=t+s-1} \mid y_1^{u-1}, h_1^T) = \qquad (7)$$
$$\begin{cases} \dfrac{1 - \sum_{\tau=t_{s-1}}^{t} p(\tau \mid t_1^{s-1}, a_1^{s-1}, h_1^T)}{1 - \sum_{\tau=t_{s-1}}^{t-1} p(\tau \mid t_1^{s-1}, a_1^{s-1}, h_1^T)} & \text{iff } y_u = \epsilon, \\[2ex] p(a_s \mid a_1^{s-1}, t_1^s, h_1^T) \cdot \left(1 - q(\epsilon \mid y_1^{u-1}, h_1^T)\right) & \text{otherwise} \end{cases}$$

with the number of segments s, segment labels a_1^s and corresponding segment boundaries t_1^s defined as follows:

$$s = \left| \{u' \in \{1, \dots, u-1\} : y_{u'} \in V\} \right| + 1,$$
$$t_{s'} = \min \{t \in \{t_{s'-1}, \dots, T\} : y_{t+s'-1} \in V\},$$
$$a_{s'} = y_{t_{s'}+s'-1} \quad \forall s' = 1, \dots, s-1.$$

Here ϵ is again regarded as segment continuation, but the final blank transition at step U is regarded as sentence end label $\#$ in this case. Substituting Eq. (7) into the transducer model defined in Eq. (3) leads to the segmental model defined in Eq. (2). This shows that using Eq. (7), also any segmental model can be rewritten into a transducer model providing an identical label posterior distribution. Therefore, both segmental model and transducer model are equivalent and provide the same modeling strength.

2.5. Special case: strict monotonicity

An additional strict monotonicity constraint with a minimum segment length of one frame can be simply adopted by modifying the segment boundary condition to $t_{s-1} < t_s \leq T$ and $t_0 \equiv 0$. This can be easily applied into the aforementioned interrelation between segmental model and transducer model by adding the additional constraint:

$$p(t_s = t_{s-1} \mid a_1^{s-1}, t_1^{s-1}, h_1^T) = 0$$

which effectively leads to:

$$q(\epsilon \mid y_1^{u-1}, h_1^T) = 1 \text{ for } y_{u-1} \in V.$$

This corresponds to time-synchronous transducer models [6, 7, 10], where the RNN-T vertical transition is replaced with a diagonal transition, i.e. $u = t$ and $U = T$. With this time-synchronous label topology, the given interrelation is still valid. Therefore, such transducer model with strict monotonicity is just one special case and is also equivalently powerful as a segmental model.

2.6. Search and pruning

This model equivalence indicates that we can apply the same transducer model as a segmental model in decoding. In general, the final best output sequence can be decided as:

$$x_1^{T'} \to \tilde{a}_1^{\tilde{S}} = \arg\max_{a_1^S, S} \ p_{\text{LM}}^\lambda(a_1^S) \cdot p(a_1^S \mid h_1^T) \qquad (8)$$

where the log-linear combination with an external language model (LM) p_{LM} using scale λ is optional. Using the Viterbi approximation, Eq. (8) for the transducer model can be further written as:

$$x_1^{T'} \to \tilde{a}_1^{\tilde{S}} = \arg\max_{a_1^S, S} \ p_{\text{LM}}^\lambda(a_1^S) \cdot \max_{y_1^U : a_1^S} p(y_1^U \mid h_1^T) \qquad (9)$$

and for the segmental model as:

$$x_1^{T'} \to \tilde{a}_1^{\tilde{S}} = \arg\max_{a_1^S, S} \; p_{\text{LM}}^\lambda(a_1^S) \cdot \max_{t_1^S} p(a_1^S, t_1^S \mid h_1^T) \qquad (10)$$

Ideally, with an identical label posterior distribution and unrestricted decoding conditions, both Eq. (9) and Eq. (10) should reveal the same optimal output sequence with the same probability.

2.6.1. Time-synchronous vs. label-synchronous search

Although both approaches are equivalent, they result in different search behavior in decoding. More precisely, time-synchronous search is usually used for transducer models [5, 7, 22], where hypotheses y_1^u are expanded per time frame t. Other variants such as alignment-length synchronous decoding [23] can also be applied for decoding transducer models, which are not investigated in this work. On the other hand, segmental models suggest label-synchronous search, where hypotheses (a_1^s, t_1^s) are expanded per segment s. This leads to a quadratic cost of search to hypothesize both labels and segment boundaries. Additionally, one can also decompose (a_1^s, t_1^s) to perform search on expansions of t before expanding a or vice versa.

2.6.2. Pruning

Pruning can also be an issue for decoding with such re-interpreted models. A common pruning method applied after each hypotheses expansion is the score-based pruning, where score refers to the negative logarithm of probability. With score-based pruning, hypotheses are pruned away if their score difference to the current best is more than a predefined threshold Q_{prune}. Another simple and common pruning method is to use a fixed beam size B. Based on score, only the best B hypotheses at each expansion step are kept for further search.

While these pruning methods are well investigated for time-synchronous search, they may cause more search error for label-synchronous search in this case. Based on Eqs. (4) and (5), sequence hypotheses of the same length might cover completely different number of encoder frames. This variation of temporal contribution can lead to an unreliable score comparison between very short and long segment hypotheses before reaching T. Thus, search can be expected to become more sensitive to pruning. The concrete effect can also vary among different settings such as subsampling and label unit choice.

3. Experiments

3.1. Phoneme-based Transducer on TED-LIUM-v2

One setup of our experiments is done on the 2nd release of the TED-LIUM corpus (TED-LIUM-v2) [24]. We use the same phoneme-based transducer model from [10], which has the strict monotonicity constraint as described in Section 2.5. Additionally, the model assumes a first-order dependency which still fits in the equivalence transformation shown in Section 2.3. This simplification largely reduces the computation complexity and allows us to investigate the score-based pruning with larger Q_{prune}. We use Eq. (9) and Eq. (10) for time-synchronous and label-synchronous search, respectively. The same 4-gram word-level LM and scale λ as in [10] are used for all recognition.

In this setup, both time-synchronous and label-synchronous decoders are implemented with the RWTH ASR toolkit [25]. The former is performed in the standard way where hypotheses expansion and score-based pruning are applied at each

Table 1: *WER comparison of time-sync. and label-sync. search using the same phoneme-based transducer model under different pruning threshold; Evaluation on the dev set of TED-LIUM-v2; And percentage of utterances where both search generate the same transcription as well as same transcription with same score*

Q_{prune}	TED-LIUM-v2 dev WER [%]		Utterance [%]	
	time-sync.	label-sync.	same-trans.	same-score
4	8.6	26.4		
6	7.5	20.9	-	-
8	7.2	17.5		
10	7.1	15.7	30.2	26.0
12	7.1	14.4	36.7	33.1
14	7.1	13.5	42.4	37.5
20	7.1	12.4	55.8	51.9

time step. The label-synchronous search is performed on full-segment expansion. Namely, for each partial path (a_1^{s-1}, t_1^{s-1}) from step $s-1$, we hypothesize (a_s, t_s) jointly at step s and compute score for this full segment. Then score-based pruning is applied among all new path hypotheses (a_1^s, t_1^s). Ended hypotheses, i.e. $t_s = T$, are kept separately for final decision without interfering further search among other paths.

The word error rate (WER) results with various pruning threshold Q_{prune} are shown in the first three columns of Table 1. For small Q_{prune}, label-synchronous search suffers a much larger degradation than time-synchronous search, which coincides with the pruning sensitivity mentioned in Section 2.6.2. With an increasing Q_{prune}, the performance of label-synchronous search gets better and better. However, due to the smooth distribution of the underlying phoneme-based transducer model, the number of hypotheses explode quickly. Besides, even with $Q_{\text{prune}} = 20$, i.e. a magnitude of 10^9 in the probability domain, it still does not reach the same WER as time-synchronous search. Although the tendency is clear to infer that with further increasing Q_{prune}, both search will eventually give the same performance. Unfortunately, this leads to a dramatic increase of memory and time for decoding, which can not be performed in this work due to hardware limitation. Here the simple label-synchronous search applied has a much worse efficiency for the originally transducer model. It also indicates the necessity of a more suitable pruning method to match the nature of the model.

To further have some insights on the equivalence perspective, we check the number of utterances where both decodings generate the same transcription (same-trans.) as well as the same transcription with same score (same-score) under a numerical tolerance of 0.0001. This is done for $Q_{\text{prune}} \geq 10$ where time-synchronous search output already saturates at the optimum. The results are shown in the last two columns of Table 1, presented as percentage of the total number of utterances in the dev set. One clear evidence here is the large increment of utterances where both decodings generate the same transcription with exactly the same score. We believe that with further increasing Q_{prune}, this number will eventually approach 100% as suggested by the model equivalence shown in Section 2.

3.2. BPE-based Transducer on Switchboard

We also perform experiments on the Switchboard corpus [26] with the byte pair encoding (BPE) [27] subword-based transducer model from [9]. This model uses full context dependency and the time-synchronous label topology as described in

Table 2: *WER comparison of time-sync. and label-sync. search using the same BPE-based transducer model under different beam sizes; Evaluation on the Switchboard Hub5'00 set; Detailed application of search and beam sizes see Section 3.2; Increasing the beam size further did not improve the result in any case*

Search	B_t	B	Hub5'00 WER [%]
time-sync.	-	1	14.0
	-	12	13.8
label-sync.	1	1	14.8
	4	4	14.2
	4	12	14.0
	100	12	14.1
	100	32	14.1
	500	12	14.1

Section 2.5. In this setup, we apply simple beam search and no additional LM is used. Both time-synchronous and label-synchronous decoders are implemented purely in RETURNN [28] as a batched fully GPU-based beam search decoder. All the code and configuration files are published [1].

For time-synchronous search, hypotheses are expanded at each time frame and a beam size B is applied for pruning. The label-synchronous search was originally implemented for our hard monotonic latent attention model [20]. As described in Section 2.6.1, this search is performed on expansions of t first before expanding a. More precisely, at search step s given the B partial path hypotheses $(a_1^{s-1}, t_1^{s-1})_{b=1,...,B}$ from step $s-1$, we firstly hypothesize t_s only and update the score for each $(a_1^{s-1}, t_1^s)_b$ based on Eq. (4). Then for each b, we apply a beam B_t to select the top position hypotheses $(a_1^{s-1}, t_1^s)_b$ based on score, which effectively leads to a total of $B \cdot B_t$ remaining (a_1^{s-1}, t_1^s). Then we expand segment label a_s for each of the remaining (a_1^{s-1}, t_1^s) and update the score for each new hypothesis (a_1^s, t_1^s) based on Eq. (5). Finally, the top-scored B hypotheses (a_1^s, t_1^s) are kept for the next search step. Here the b-individual position hypotheses pruning with beam B_t is to avoid search errors based on duration probability only without the label probability yet.

The WER results with different beam settings are shown in Table 2. In this setup, the performance of time-synchronous and label-synchronous search are much closer, although the former is still slightly better. This much smaller difference is probably due to two reasons: Firstly, a high subsampling factor of 6 is applied so that label segments become short in general. Secondly, the model produces a much sharper distribution over the BPE units. Both aspects make the label-synchronous search less sensitive to pruning. By further increasing the beam sizes B_t and B, no improvement is obtained, while the GPU memory is quickly exceeded.

4. Conclusions

In this work, we proved the equivalence of transducer models and segmental models. We showed transformation equations that allow to rewrite any transducer model into a segmental model and vice versa. This covers both the standard RNN-T topology and the additional strict monotonicity constraint as a special case. Based on this equivalence, both time-

---

[1] https://github.com/rwth-i6/returnn-experiments/tree/master/2021-segmental-transducer-equivalence

synchronous and label-synchronous search can be applied for beam search decoding using either model. This is experimentally investigated with the phoneme-based transducer model on TED-LIUM-v2 and the BPE-based transducer model on Switchboard. Initial experiments show that standard score-based and beam size-based pruning techniques are insufficient for optimal decoding in a label-synchronous fashion. Although the equivalence shows that both transducer models and segmental models are equally powerful, a time-synchronous decoding paradigm currently shows more efficient pruning behavior for inherently transducer models.

Additionally, the transformation equations presented would also allow for time-synchronous decoding of originally segmental models. The analytical investigation of decoding with both transducer and segmental models might inspire future work on improving search and pruning approaches in a more general framework.

5. Acknowledgements

6. References

[1] H. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*. Springer, 1994, vol. 247.

[2] A. Zeyer, P. Doetsch, P. Voigtlaender, R. Schlüter, and H. Ney, "A comprehensive study of deep bidirectional LSTM RNNs for acoustic modeling in speech recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, New Orleans, LA, USA, Mar. 2017, pp. 2462–2466.

[3] T. Raissi, E. Beck, R. Schlüter, and H. Ney, "Context-Dependent Acoustic Modeling without Explicit Phone Clustering," in *Proc. Interspeech*, 2020.

[4] A. Graves, S. Fernández, F. J. Gomez, and J. Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," in *Proc. Int. Conf. on Machine Learning (ICML)*, 2006, pp. 369–376.

[5] A. Graves, "Sequence Transduction with Recurrent Neural Networks," 2012, https://arxiv.org/abs/1211.3711.

[6] H. Sak, M. Shannon, K. Rao, and F. Beaufays, "Recurrent Neural Aligner: An Encoder-Decoder Neural Network Model for Sequence to Sequence Mapping," in *Proc. Interspeech*, F. Lacerda, Ed., 2017.

[7] A. Tripathi, H. Lu, H. Sak, and H. Soltau, "Monotonic Recurrent Neural Network Transducer and Decoding Strategies," in *IEEE ASRU*, 2019, pp. 944–948.

[8] E. Variani, D. Rybach, C. Allauzen, and M. Riley, "Hybrid Autoregressive Transducer (HAT)," in *Proc. ICASSP*, 2020, pp. 6139–6143.

[9] A. Zeyer, A. Merboldt, R. Schlüter, and H. Ney, "A New Training Pipeline for an Improved Neural Transducer," in *Proc. Interspeech*, 2020.

[10] W. Zhou, S. Berger, R. Schlüter, and H. Ney, "Phoneme Based Neural Transducer for Large Vocabulary Speech Recognition," in *Proc. ICASSP*, 2021.

[11] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[12] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[13] A. Zeyer, K. Irie, R. Schlüter, and H. Ney, "Improved training of end-to-end attention models for speech recognition," in *Proc. Interspeech*, 2018, pp. 7–11.

[14] 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," in *Proc. Interspeech 2019*, 2019, pp. 2613–2617.

[15] Z. Tüske, G. Saon, K. Audhkhasi, and B. Kingsbury, "Single headed attention based sequence-to-sequence model for state-of-the-art results on Switchboard," in *Proc. Interspeech 2020*, 2020, pp. 551–555.

[16] M. Ostendorf, V. V. Digalakis, and O. A. Kimball, "From HMM's to segment models: A unified view of stochastic modeling for speech recognition," *IEEE Transactions on speech and audio processing*, vol. 4, no. 5, pp. 360–378, 1996.

[17] G. Zweig and P. Nguyen, "A segmental CRF approach to large vocabulary continuous speech recognition," in *ASRU*, 2009, p. 35.

[18] E. Beck, M. Hannemann, P. Doetsch, R. Schlüter, and H. Ney, "Segmental encoder-decoder models for large vocabulary automatic speech recognition," in *Interspeech*, Hyderabad, India, Sep. 2018.

[19] E. Beck, A. Zeyer, P. Doetsch, A. Merboldt, R. Schlüter, and H. Ney, "Sequence modeling and alignment for LVCSR-systems," in *ITG Conference on Speech Communication*, Oldenburg, Oct. 2018.

[20] A. Zeyer, R. Schlüter, and H. Ney, "A study of latent monotonic attention variants," Preprint ArXiv:2103.16710, 2021.

[21] E. Beck, M. Hannemann, P. Doetsch, R. Schlüter, and H. Ney, "Segmental Encoder-Decoder Models for Large Vocabulary Automatic Speech Recognition," in *Proc. Interspeech*, 2018.

[22] J. Kim and Y. Lee, "Accelerating RNN Transducer Inference via One-Step Constrained Beam Search," 2020, https://arxiv.org/abs/2002.03577.

[23] G. Saon, Z. Tüske, and K. Audhkhasi, "Alignment-length synchronous decoding for RNN transducer," in *Proc. ICASSP*, 2020, pp. 7804–7808.

[24] A. Rousseau, P. Deléglise, and Y. Estève, "Enhancing the TED-LIUM Corpus with Selected Data for Language Modeling and More TED Talks," in *Proc. LREC*, 2014, pp. 3935–3939.

[25] S. Wiesler, A. Richard, P. Golik, R. Schlüter, and H. Ney, "RASR/NN: The RWTH Neural Network Toolkit for Speech Recognition," in *Proc. ICASSP*, 2014, pp. 3281–3285.

[26] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "SWITCHBOARD: Telephone Speech Corpus for Research and Development," in *Proc. ICASSP*, vol. 1, 1992, pp. 517–520.

[27] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *ACL*, Berlin, Germany, August 2016, pp. 1715–1725.

[28] P. Doetsch, A. Zeyer, P. Voigtlaender, I. Kulikov, R. Schlüter, and H. Ney, "RETURNN: the RWTH extensible training framework for universal recurrent neural networks," in *Proc. ICASSP*. IEEE, 2017, pp. 5345–5349.