REGARDING TOPOLOGY AND ADAPTABILITY IN DIFFERENTIABLE WFST-BASED E2E ASR

Zeyu Zhao[†]*, Pinzhen Chen*[‡]*, Peter Bell*[†]

[†]Centre for Speech Technology Research, University of Edinburgh, UK
[‡]School of Informatics, University of Edinburgh, UK

ABSTRACT

The adaptability of End-to-End (E2E) Automatic Speech Recognition (ASR) models across diverse datasets remains a challenge, often attributed to acoustic model (AM) generalisability and the internal language model (ILM) mismatch. This study delves into the impact of topology on adaptability in Differentiable WFST-based ASR. Through evaluations on various ASR corpora, we discern a significant influence of topology on adaptability. Notably, Connectionist Temporal Classification's performance diminishes with substantial acoustic feature deviations from its training set. Additionally, we confirm that the internal language models within these topologies are sufficiently weak, indicating that acoustic model generalisability is the primary factor influencing adaptability.

Index Terms— E2E ASR, Differentiable WFST, Topology, Internal Language Model, Acoustic Modelling

1. INTRODUCTION

Adapting an End-to-End (E2E) Automatic Speech Recognition (ASR) model from one dataset to another presents significant challenges, primarily due to the acoustic and linguistic feature mismatches between the source and target domains [1–5]. In E2E ASR, regardless of the methodologies employed [6–8], there is a consistent effort to directly model the posterior $p(Y|X)$, where X denotes the speech input and Y represents the target sequence. According to Bayes' rule,

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)}, \tag{1}$$

in an E2E ASR model, two primary components are implicitly present: an Acoustic Model (AM), represented as $p(X|Y)$ and an "Internal" Language Model (ILM), denoted as $p(Y)$. A significant portion of research has centred on domain adaptation placing a greater emphasis on the ILM [1–3, 5]. This is particularly crucial as a mismatched ILM can lead to suboptimal performance on evaluation data [1–3, 5]. Notably, much of this research has been grounded in the frameworks of the Recurrent Neural Network Transducer (RNN-T) [7] and the Attention-based Encoder Decoder (AED) [8]. This might be attributed to the inherent language model-like structures in both: the prediction network in RNN-T and the decoder in AED. While the focus on ILM is substantial, there is also a body of work dedicated to the AM aspect [4, 9], exploring the acoustic modelling and adaptive capabilities of Connectionist Temporal Classification (CTC) [6] and Lattice-Free Maximum Mutual Information (LF-MMI) [10, 11]. However, we posit that for effective adaptation of an E2E ASR model to unfamiliar speech data, both AM and ILM components warrant consideration.

CTC inherently adopts a one-state topology as highlighted in [12]. The topology's role in E2E ASR is crucial for acoustic modelling, as it defines how modelling units are constructed from neural network (NN) outputs. While Hidden Markov Model (HMM) ASR underwent extensive topology exploration, culminating in the Bakis topology [13], CTC gained popularity without in-depth evaluation of its inherent topology. We observe that CTC's simplistic one-state topology struggles to generalise across varied acoustic conditions. This raises questions about the topology's effect on AM generalisability and its subsequent impact on adaptability. Besides, we are also curious about whether topology may influence ILM. Fortunately, Differentiable Weighted Finite-State Transducer (DWFST) [14] enables us to explore various topology configurations. This study examines the role of topology in DWFST-based E2E ASR, focusing on adaptability in both AM and ILM.

2. METHOD

2.1. Training

Given input speech X in the form of a sequence of feature vectors or raw waveform, by applying an acoustic encoder, we obtain a sequence of hidden features or high-level representations. Next, we apply a (set of) linear layer(s) with a final softmax activation to obtain a probability distribution over each token π_t at time step t, $p_t(\pi_t|X)$, $t = 1, 2, \ldots, T$.

The posterior probability $p(Y|X)$ is modelled as

$$p(Y|X) = \frac{\sum_{\pi \in \Pi(Y; \boldsymbol{T} \circ \boldsymbol{L})} p(\pi|X)}{\sum_{Y'} \sum_{\pi \in \Pi(Y'; \boldsymbol{T} \circ \boldsymbol{L})} p(\pi|X)}, \tag{2}$$

where in the denominator, Y' represents any arbitrary word sequence, and $\Pi(Y; \boldsymbol{F})$ denotes the set of all the token sequences ("paths") corresponding to the output label sequence Y in the FST $\boldsymbol{F}$. $\boldsymbol{T}$ and $\boldsymbol{L}$ denote the Token FST and the Lexicon FST, respectively, and $\circ$ denotes the composition of WFSTs [12, 14, 15]. We apply the conditional independence assumption and thus

$$p(\pi|X) = \prod_{t=1}^{T} p_t(\pi_t|X). \tag{3}$$

We compute the numerator and the denominator in eq. (2) with DWFST. Specifically, for the numerator,

$$\sum_{\pi \in \Pi(Y; \boldsymbol{T} \circ \boldsymbol{L})} p(\pi|X) = \mathrm{TotalScore}(\boldsymbol{E} \circ \boldsymbol{S}^{\mathrm{trn}}), \tag{4}$$

where $\mathrm{TotalScore}$ denotes the total score operation, and $\boldsymbol{E}$ is the Emission FST constructed from the neural network model's output. $\boldsymbol{S}^{\mathrm{trn}}$ is the training graph derived from the target sequence Y,

$$\boldsymbol{S}^{\mathrm{trn}} = \boldsymbol{T} \circ (\boldsymbol{L} \circ \boldsymbol{W}), \tag{5}$$

where $\boldsymbol{W}$ is a linear Word FST with the word sequence Y as input and output labels [14]. We should emphasise that the term *topology* discussed in this paper is the topology of $\boldsymbol{T}$ which describes how to form a modelling unit (e.g., phoneme, character, or byte pair encoding [16]) from NN's outputs, so it is crucial to the AM generalisability.

Our previous work [17] has underscored the importance of normalisation (specifically of the denominator term in eq. (2)) for achieving training convergence. However, it is unrealistic to create a training graph as eq. (5) for every possible word sequence. Thus, to compute the denominator, given $\boldsymbol{E}$,we solely take into account the paths that are acceptable for $\boldsymbol{T}$, so we have

$$\sum_{Y'} \sum_{\pi \in \Pi(Y'; \boldsymbol{T} \circ \boldsymbol{L})} p(\pi|X) \approx \mathrm{TotalScore}(\boldsymbol{E} \circ \boldsymbol{T}). \tag{6}$$

We note that when $\boldsymbol{T}$ is CTC topology, as shown in fig. 1a, any arbitrary token sequence can be accepted by $\boldsymbol{T}$, so the denominator is always one, resulting in

$$p(Y|X) = \sum_{\pi \in \Pi(Y; \boldsymbol{T}_{\mathrm{CTC}} \circ \boldsymbol{L})} p(\pi|X), \tag{7}$$

which is equivalent to the CTC loss function [6]. Except for a few topologies, e.g. CTC, not all paths in E can be accepted by $\boldsymbol{T}$. This leads to a denominator not equal to one.

2.2. Decoding

We construct the decoding graph as

$$\boldsymbol{S}^{\mathrm{dec}} = \boldsymbol{T} \circ (\boldsymbol{L} \circ \boldsymbol{G}), \tag{8}$$

where $\boldsymbol{G}$ is a grammar FST, often obtained from an n-gram language model [12]. For simplicity, operations such as determinisation and minimisation [15] have been omitted in the equation. With the output of the neural network model and the decoding graph, we apply the Viterbi decoder [18, 19] to get the best word sequence

$$W^* = \arg\max_W [\log p(W) + \alpha \max_{\pi \in \Pi(W; \boldsymbol{S}^{\mathrm{dec}})} \log p(\pi|X)], \tag{9}$$

where $p(W)$ is determined by the language model, which is encoded in $\boldsymbol{S}^{\mathrm{dec}}$ as transition weights, and α is the acoustic weight.

3. EXPERIMENTS

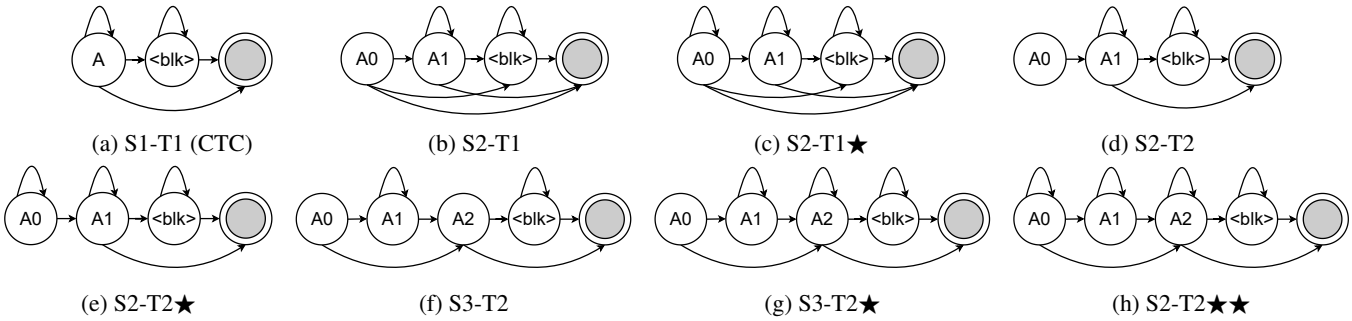
3.1. Topologies

For a comprehensive investigation, we examine eight various topologies, as illustrated in fig. 1. **S1-T1** is the standard CTC topology and serves as our baseline. In this configuration, each phone corresponds to a single token, and a mandatory blank label acts as a separator for consecutive repeated phones. **S2-T1** is inspired by our previous work [17] to enhance the modelling capabilities by introducing an additional state for each phone. This topology comprises a primary state without a self-loop and an optional and skippable secondary state. **S2-T1★** is an adaptation of S2-T1, distinguished by the inclusion of a self-loop on the first state. To delve deeper into the implications of frame absorption, we present five more topologies. **S2-T2★** extends S1-T1 by incorporating an additional state equipped with a self-loop. **S2-T2** is a variant of S2-T2★ without the self-loop. We then introduce **S3-T2** by retaining the same minimum traversal frame and integrating an extra state. Finally, **S3-T2★** and **S3-T2★★** are variants of S3-T2, with modifications in the self-loop and skip transition configurations respectively.

For consistency across comparisons, all the topologies mentioned above include a shared optional blank token to keep a fair comparison with the CTC topology.

3.2. Datasets

To assess the AM generalisability, we train the models with various topologies on Wall Street Journal (WSJ) dataset, which offers around 80 hours of speech data, and evaluate them on various corpora. For evaluation, we utilise multiple datasets: WSJ(eval92, 0.7 hours), LibriSpeech (test-other, 5.3 hours), Tedlium3 (2.6 hours), and AMI (8.6 hours). While both LibriSpeech and WSJ feature professionally recorded speech in controlled settings, Tedlium3, sourced from TED talks, offers a less pristine audio quality. AMI, capturing audio in meeting scenarios with overlapping speech, stands out as the most challenging dataset in our evaluation. The above evaluation sets allow us to assess the adaptive capabilities and robustness of the different topologies in various scenarios.



(a) S1-T1 (CTC) (b) S2-T1 (c) S2-T1★ (d) S2-T2

(e) S2-T2★ (f) S3-T2 (g) S3-T2★ (h) S2-T2★★

Fig. 1: Illustrations of the 8 topologies investigated in this paper. <blk> denotes the shared optional blank label. Note that in some cases the blank label is unskippable for CTC but we omit it for simplicity. Sx-Ty means there are x states for each phone, the minimum traversal frame is y, and one ★ means one more self-loop is added.

3.3. Data Clustering

Apart from performing adaptation between public datasets, we create an additional controlled experiment on two subsets of recordings from Tedlium3 training set (2351 recordings in total). We aim to, in an unsupervised manner, create two subsets that dramatically vary in terms of language modelling but have consistent acoustic patterns.

We first map all Tedlium3 documents into unigram bag-of-words vectors, with words appearing less than 3 times globally turned into <unk>. Then term frequency-inverse document frequency (TF-IDF) weighting is applied on a logarithmic scale with base 10 [20]. With these vectors, we perform k-means clustering [21]; having an unknown number of latent clusters, we loop k from 2 to 15 and inspect the two largest clusters as potential subsets. Our constraints on cluster selection are as follows: 1) minimising the difference between the number of recordings in the two clusters to achieve two balanced subsets; 2) maximising the total number of recordings in the two clusters to have ample data for training; 3) maximising the perplexity of a language model trained on one subset and scored on the other subset, to maximise the language difference.

Empirically choosing k as 6, we obtain two primary subsets: C0 with 798 recordings and C1 with 719. To balance the subsets, we randomly exclude 79 from C0. Next, from the adjusted subsets, 71 recordings are randomly allocated to development (20 for each, dev0 and dev1) and test sets (51 for each, test0 and test1), leaving 648 for training (train0 and train1). Using KenLM [22], we train two 3-gram LMs on train0 and train1, and measure perplexity on the combined development and test sets. These two 3-gram LMs are also utilised later in our ASR experiments for decoding.

Table 1 shows a notable perplexity gap between C0 and C1. Given that both subsets originate from Tedlium3, they exhibit acoustic similarities, suggesting minimal challenges in acoustic adaptability. Consequently, topologies with a strong ability to learn the internal language model may struggle to adapt between the subsets.

Table 1: Perplexity of subset LMs on held-out sets.

	LM0 (train0)	LM1 (train1)
(dev0+test0)	207.92	259.29
(dev1+test1)	244.65	191.75

3.4. Settings

In our training setup, we initialise with the wav2vec 2.0 model [23] pre-trained on the expansive Libri-Light dataset with 60k hours of untranscribed data (`WAV2VEC2_LARGE_LV60K`). We fine-tune the last 12 transformer layers of the model and the output linear layers on WSJ. We employ the Adam optimizer with a learning rate of 10^{-4} for the wav2vec 2.0 model, while the output linear layers utilise the Adadelta optimizer with a learning rate of 0.9. Our data preparation follows Kaldi [24] and ESPnet [25]. With PyTorch [26] driving our neural network training, we apply k2[1] as the DWFST backend. In all experiments, we utilise 39 English phonemes as the modelling units. For the S1-T1 topology, the output units total 41 (comprising 39 phones, <unk>, and <blank>). The S2-* topologies have 81 output units, and the S3-* topologies have 121. All audio inputs are resampled to 16kHz (if not originally at this rate) and normalised [23]. To augment our dataset, we apply speed perturbation on WSJ. For decoding, we apply the Viterbi decoder from Kaldi, complemented by specific n-gram language models for each evaluation set: 4-gram for WSJ, 3-gram "tgsmall" for Librispeech, big 4-gram for Tedlium3, and a pruned 3-gram for AMI. Acoustic weights are tuned with these language models to achieve optimal performance on the corresponding development sets. We also make our code open-source.[2]

[1] https://github.com/k2-fsa/k2

[2] https://github.com/ZhaoZeyu1995/Waterfall

3.5. Results and Analysis

3.5.1. Overall Performance

Table 2 presents the performance (Word Error Rate, WER%) across different evaluation sets and topologies. On WSJ eval92 (the WSJ column in table 2), S3-T2★★ leads the pack. While absolute WER values are closely matched, the WERR of S3-T2★★ relative to the baseline S1-T1 (CTC) stands at 11.1%. On Librispeech test-other (the LS column in table 2), S2-T1★ and S3-T2★★ take the lead, with S1-T1 trailing closely, likely due to the acoustic resemblance between Librispeech and WSJ.

Table 2: Performance (WER%) on various evaluation sets with different topologies. The last column is the weighted (concerning the total length of each dataset) average WER Reduction (WERR%) compared to the baseline, S1-T1 (CTC) topology.

	WSJ	LS	TED3	AMI	WERR%
S1-T1 (CTC)	2.7	8.1	8.9	35.2	-
S2-T1	2.9	8.8	**8.1**	31.6	3.5
S2-T1★	2.6	**8.0**	8.3	33.4	4.1
S2-T2★	2.9	8.6	8.2	30.6	5.5
S2-T2	2.8	8.3	8.3	**30.2**	**7.2**
S3-T2	2.9	8.9	**8.1**	31.5	3.3
S3-T2★	2.7	9.0	8.4	32.3	1.5
S3-T2★★	**2.4**	**8.0**	8.3	32.2	6.1

On Tedlium3, S1-T1 (CTC) notably underperforms against other topologies. This limitation becomes stark on AMI, where S2-T2 achieves a WERR of 14.2% over S1-T1. The root of this disparity lies in S1-T1's approach of using a single state to represent each phoneme. This assumes consistent acoustic properties throughout the phoneme, which does not hold usually. Consequently, the model tends to identify only the most distinct phoneme segments, relegating most frames to blank labels. This phenomenon, known as the "peaky" issue in CTC [27,28], diminishes acoustic modelling capabilities and can lead to overfitting to the training set's specific acoustic conditions. On Tedlium3, S2-T1 and S3-T2 outshine other topologies, despite not leading on WSJ and Librispeech. This underscores the distinct acoustic nature of Tedlium3 compared to the former two datasets. Similarly, on AMI, S2-T2 emerges as the best, even though it does not top the charts on the other three datasets. Consequently, no single "supreme" topology excels consistently across all datasets. Yet, when evaluating the weighted (by the length of each evaluation set) average WERR against the baseline S1-T1 (CTC) topology, S2-T2 stands out, achieving an average WERR of 7.2% over CTC. We find that all the examined topologies generalise better than CTC, which implies that CTC could be the easiest option but normally not the best.

We have to admit that we have not come up with a perfect theory to explain why a specific topology can achieve the best adaptability on a dataset or to predict which topology will achieve optimal performance given a specific condition. However, we note that by changing the topology, just like changing neural network hyper-parameters or other training settings, we have another dimension to control ASR models.

Table 3: Performance (WER%) of the models trained on **train1**, evaluated on test0 and test1 with or without 3-gram LMs trained on train0 (LM0) or train1 (LM1)

Topology	test1		test0	
	noLM	+LM1	noLM	+LM0
S1-T1 (CTC)	15.8	13.2	14.4	12.2
S2-T1	15.0	12.7	14.1	12.1
S2-T1★	14.8	12.6	13.8	12.0
S2-T2★	15.6	13.0	14.2	12.1
S2-T2	15.0	12.7	13.9	12.0
S3-T2	15.2	12.8	14.0	12.1
S3-T2★	15.4	12.9	13.8	12.0
S3-T2★★	14.9	12.7	14.2	12.1

3.5.2. Effect of Internal Language Models

As outlined in section 1, adaptability in an E2E ASR model is influenced by two factors: AM generalisability and ILM mismatch. Here, we would like to verify that topology mainly affects adaptability by influencing AM generalisability. As detailed in section 3.3, we derived two Tedlium3 subsets with distinct language patterns but nearly identical acoustic features. If a topology drives the model to learn a strong ILM from one subset, it may struggle to adapt to the other. Therefore, a topology's underperformance in the results is indicative of its inclination to learn a strong ILM. Due to space constraints, we display the performance of models trained on train1 in table 3; results with train0 are similarly aligned. With or without the corresponding external LM, models trained on train1 excel on test0 over test1. The ILM's impact seems minimal across all the examined topologies. Thus, we see no signs of a dominant ILM in any topology, indicating AM generalisability as the primary adaptability factor in DWFST-based E2E ASR. Note that our findings are based on phoneme modelling units, and outcomes might differ with other units.

4. CONCLUSION

This study explored the role of topology in DWFST-based E2E ASR adaptability. We found that the topology of ASR models influences the adaptability. Evaluations across multiple ASR corpora highlighted that topology affects adaptability by influencing acoustic modelling power, as no dominant internal language model was found. Besides, the popular CTC topology struggled with significant acoustic deviations from its training data. While our findings are anchored on phonemes as the primary modelling units, future work can explore diverse units to enhance these insights.

5. REFERENCES

[1] Zhong Meng et al., "Internal Language Model Training for Domain-Adaptive End-To-End Speech Recognition," in *ICASSP*, June 2021, pp. 7338–7342.

[2] Zhong Meng et al., "Internal Language Model Estimation for Domain-Adaptive End-to-End Speech Recognition," in *SLT*, 2021, pp. 243–250.

[3] Zhong Meng et al., "Internal Language Model Adaptation with Text-Only Data for End-to-End Speech Recognition," in *Interspeech 2022*. Sept. 2022, pp. 2608–2612, ISCA.

[4] Apoorv Vyas et al., "Comparing CTC and LFMMI for Out-of-Domain Adaptation of wav2vec 2.0 Acoustic Model," in *Interspeech 2021*. Aug. 2021, pp. 2861–2865, ISCA.

[5] Janne Pylkkönen et al., "Fast Text-Only Domain Adaptation of RNN-Transducer Prediction Network," in *Interspeech 2021*. Aug. 2021, pp. 1882–1886, ISCA.

[6] Alex Graves et al., "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006, pp. 369–376.

[7] Alex Graves, "Sequence Transduction with Recurrent Neural Networks," *arXiv:1211.3711 [cs, stat]*, Nov. 2012.

[8] William Chan et al., "Listen, Attend and Spell," *arXiv:1508.01211 [cs, stat]*, Aug. 2015.

[9] Sibo Tong et al., "Cross-lingual adaptation of a CTC-based multilingual acoustic model," *Speech Communication*, vol. 104, pp. 39–46, Nov. 2018.

[10] Hossein Hadian et al., "Flat-Start Single-Stage Discriminatively Trained HMM-Based Models for ASR," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 11, pp. 1949–1961, 2018.

[11] Hossein Hadian et al., "End-to-end Speech Recognition Using Lattice-free MMI," in *Interspeech 2018*. Sept. 2018, pp. 12–16, ISCA.

[12] Y. Miao et al., "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *ASRU*, Dec. 2015, pp. 167–174.

[13] R. Bakis, "Continuous speech recognition via centisecond acoustic states," *The Journal of the Acoustical Society of America*, vol. 59, no. S1, pp. S97–S97, Apr. 1976.

[14] Awni Hannun et al., "Differentiable Weighted Finite-State Transducers," *arXiv:2010.01003 [cs, stat]*, Oct. 2020.

[15] Mehryar Mohri et al., "Speech Recognition with Weighted Finite-State Transducers," in *Springer Handbook of Speech Processing*, Jacob Benesty et al., Eds., pp. 559–584. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[16] Taku Kudo and John 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*, Eduardo Blanco and Wei Lu, Eds., Brussels, Belgium, Nov. 2018, pp. 66–71, Association for Computational Linguistics.

[17] Zeyu Zhao and Peter Bell, "Investigating Sequence-Level Normalisation For CTC-Like End-to-End ASR," in *ICASSP*, 2022, pp. 7792–7796.

[18] Daniel Povey et al., "Generating exact lattices in the WFST framework," in *ICASSP*, Mar. 2012, pp. 4213–4216.

[19] Lawrence R Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *PROCEEDINGS OF THE IEEE*, vol. 77, no. 2, pp. 30, 1989.

[20] Karen Sparck Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of documentation*, vol. 28, no. 1, pp. 11–21, 1972.

[21] Stuart Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, 1982.

[22] Kenneth Heafield, "KenLM: Faster and smaller language model queries," in *Proceedings of the Sixth Workshop on Statistical Machine Translation*, Edinburgh, Scotland, July 2011, pp. 187–197, Association for Computational Linguistics.

[23] Alexei Baevski et al., "Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," in *Advances in Neural Information Processing Systems*. 2020, vol. 33, pp. 12449–12460, Curran Associates, Inc.

[24] Daniel Povey et al., "The Kaldi speech recognition toolkit," in *ASRU*. 2011, number CONF, IEEE Signal Processing Society.

[25] Shinji Watanabe et al., "ESPnet: End-to-End Speech Processing Toolkit," in *Interspeech 2018*. Sept. 2018, pp. 2207–2211, ISCA.

[26] Adam Paszke et al., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Advances in Neural Information Processing Systems*. 2019, vol. 32, Curran Associates, Inc.

[27] Albert Zeyer et al., "Why does CTC result in peaky behavior?," June 2021.

[28] Zeyu Zhao and Peter Bell, "Regarding Topology and Variant Frame Rates for Differentiable WFST-based End-to-End ASR," in *INTERSPEECH 2023*. Aug. 2023, pp. 4903–4907, ISCA.