CRF-BASED SINGLE-STAGE ACOUSTIC MODELING WITH CTC TOPOLOGY

Hongyu Xiang, Zhijian Ou

Speech Processing and Machine Intelligence (SPMI) Lab, Tsinghua University, China
xianghy16@mails.tsinghua.edu.cn, ozj@tsinghua.edu.cn

ABSTRACT

In this paper, we develop conditional random field (CRF) based single-stage (SS) acoustic modeling with connectionist temporal classification (CTC) inspired state topology, which is called CTC-CRF for short. CTC-CRF is conceptually simple, which basically implements a CRF layer on top of features generated by the bottom neural network with the special state topology. Like SS-LF-MMI (lattice-free maximum-mutual-information), CTC-CRFs can be trained from scratch (flat-start), eliminating GMM-HMM pre-training and tree-building. Evaluation experiments are conducted on the WSJ, Switchboard and Librispeech datasets. In a head-to-head comparison, the CTC-CRF model using simple Bidirectional LSTMs consistently outperforms the strong SS-LF-MMI, across all the three benchmarking datasets and in both cases of mono-phones and mono-chars. Additionally, CTC-CRFs avoid some ad-hoc operation in SS-LF-MMI.

Index Terms— CRF, CTC, single-stage

1. INTRODUCTION

In recent years, deep neural networks (DNNs) of various different network architectures have advanced the state-of-the-art on large scale automatic speech recognition (ASR) tasks. Initially, DNN-HMM hybrid systems were developed [1], which rely on GMM-HMM models to obtain initial alignments and state-tying decision trees for context-dependent (CD) phone modeling. Recently, there are increasing interests in building end-to-end ASR systems. In contrast to previous multistage ASR systems, an end-to-end model is of single-stage, which eliminates GMM-HMM pre-training and tree-building, and can be trained from scratch (flat-start) [2, 3, 4]. In a more strict sense, only those single-stage models that remove the need for a pronunciation lexicon and, even further, train the acoustic and language models jointly rather than separately, are called end-to-end [5, 6]. Three main classes of end-to-end models are based on Connectionist Temporal Classification (CTC) [2, 7], attention based Seq2Seq [6, 8] and RNN-transducer (RNN-T) [9, 10] respectively.

Though the simplicity of "end-to-end" is appealing, pure data-driven end-to-end models are data-hungry, which require tens of thousands hours of labeled speech to achieve state-of-the-art results [8]. Note that in most application scenarios, pronunciation lexicon and large-scale text corpus for language modeling are readily available. In this paper, we are interested in advancing single-stage acoustic models, which use a separate language model (LM) with or without a pronunciation lexicon.

It is shown in [4] that single-stage (SS) lattice-free maximum-mutual-information (LF-MMI) with tree-free CD modeling technique achieves 10 to 25% relative WER reduction compared to other end-to-end methods (CTC, Seq2Seq, RNN-T) on well-known databases, including 80-h WSJ, 300-h Switchboard and 2000-h Fisher+Switchboard datasets. Basically, SS-LF-MMI is cast as MMI-based discriminative training of a HMM (generative model) with pseudo state-likelihoods calculated by the bottom neural network with fixed state-transition probabilities. The labels could be phones or characters, context-independent (CI) or CD, yielding four cases. In any case, 2-state HMM topology is used (with best performance reported in [3, 4]) and a silence label is included.

In contrast, in this paper, we develop conditional random field (CRF) [11] based single-stage acoustic modeling with CTC topology, which is called CTC-CRF for short. A CRF is a discriminative model by itself, which directly defines a conditional distribution $p(\boldsymbol{\pi}|\boldsymbol{x})$ for the state sequence $\boldsymbol{\pi} \triangleq \pi_1, \cdots \pi_T$ given the observation sequence $\boldsymbol{x} \triangleq x_1, \cdots x_T$. By introducing a blank label, the CTC topology defines a mapping $\mathcal{B}(\cdot)$ that specifies how a state sequence $\boldsymbol{\pi}$ maps to a label sequence $\boldsymbol{l} \triangleq l_1, \cdots l_L$, i.e. removing consecutive repetitive labels and blanks. Then the acoustic-to-label probability which underlies the CTC-CRF acoustic model is naturally given as follows:

$$p(\boldsymbol{l}|\boldsymbol{x}) = \sum_{\boldsymbol{\pi} \in \mathcal{B}^{-1}(\boldsymbol{l})} p(\boldsymbol{\pi}|\boldsymbol{x}) \tag{1}$$

We leave discussions about the connection and comparison of CTC-CRF with other existing models in ASR to Section 2. CTC-CRF is conceptually simple, which basically implements a CRF layer on top of features generated by the bottom neural network with the special state topology. It can be trained from scratch, namely in single-stage. Evaluation experiments are conducted on the WSJ, Switchboard and Librispeech datasets, representing different sizes of training data. In a head-to-head comparison, the CTC-CRF model using simple Bidirectional LSTMs consistently outperforms the strong SS-LF-MMI [3, 4], across all the three benchmarking datasets and in both cases of mono-phones and mono-chars. These results are very encouraging, though bi-phone/bi-char SS-LF-MMIs perform better than mono-phone/mono-char CTC-CRFs. We leave bi-phone/bi-char CTC-CRFs as further work. Additionally, CTC-CRFs avoid the ad-hoc operation of random insertions of silence phones in transcriptions for estimating the denominator LM in SS-LF-MMI.

2. RELATED WORK

In Table 1, we give a brief review of existing models in ASR, depending on state topologies, training objectives and whether the model distribution is locally or globally normalized. We differentiate HMM topology and CTC topology, though the later may be interpreted as a special HMM topology [12]. The two differ not only in the state transition structure but also in the label inventory used (which affects not only the definition of the whole state space but also the estimation of the denominator LM). As can been seen from the following

This work is supported by NSFC grant 61473168.

review and to the best of our knowledge, this paper represents the first exploration of CRFs with CTC topology.

2.1. Relation to LF-MMI

Basically, LF-MMI is cast as MMI-based discriminative training of a HMM with pseudo state-likelihoods calculated by the bottom neural network with fixed state-transition probabilities. Since HMMs are generative models, maximizing the training objective $\log p(\boldsymbol{l}|\boldsymbol{x})$ is often referred to as MMI training. Alternatively, LF-MMI can be interpreted as conditional maximum likelihood (CML) training of a CRF, which uses the neural network output as the node potential. Such equivalence is discussed in the early development of using CRFs in ASR, which use zero, first and second order features in potential definition. It is shown in [13] that MMI training of GMM-HMMs is equivalent to CML training of such CRFs on the function level. When neural networks are used to define features, it can be seen that this equivalence still holds. For the two manners - indirectly formulated as MMI training of a pseudo HMM or directly formulated as CML training of a CRF, it would be conceptually simple to adopt the later manner.

Specifically, the main differences between our CTC-CRF and SS-LF-MMI in [3, 4] are:

- The two have different state topologies, as explained in Table 1. We use the term state topology to refer not only to the state transition structure for a label but also to the label inventory.
- Since SS-LF-MMI includes the silence label, randomly inserting silence labels with probability 0.2 between the words and with probability 0.8 at the beginning and end of the sentences is used for estimating the n-gram denominator LM, which may be inaccurate. Our CTC-CRF avoids such ad-hoc operation as we do not include the silence symbol but use the blank.
- The alignments between observations and states in CTC-CRF are found to be similar to those in CTC, i.e. the alignments are usually dominated by the blank symbols and the non-blank symbols occur with spikes in their posteriors. The techniques for CTC decoding [14] can also be used for CTC-CRF to speed up decoding.

Note that besides MMI, there are other sequence discriminative training strategies, e.g. state-level minimum Bayes risk (sMBR) criterion, which has been applied to DNN-HMM hybrid systems [15] and CTC systems [16]. CTC-CRFs could also be further trained based on sMBR. But sMBR based training needs to generate denominator lattices by decoding with word-level LMs, which is more time-consuming.

2.2. Relation to CRF-based acoustic models

ASR is a sequence transduction problem in that the input and output sequences differ in lengths, and both lengths are variable. An idea in applying CRFs to ASR is to introduce a (hidden) state sequence $\boldsymbol{\pi}$ to align the label sequence $\boldsymbol{l}$ and observation sequence $\boldsymbol{x}$, and define a CRF $p(\boldsymbol{\pi}|\boldsymbol{x})$ over the (hidden) state sequence $\boldsymbol{\pi}$. As shown in Eq. (1), deriving $p(\boldsymbol{l}|\boldsymbol{x})$ based on $p(\boldsymbol{\pi}|\boldsymbol{x})$ depends on the mapping between $\boldsymbol{\pi}$ and $\boldsymbol{l}$, which is determined by the state topology that allows for different choices, e.g. CTC topology or HMM topology. This kind of hidden CRFs was explored in [17] for phone classification, using zero, first and second order features. As reviewed before, (hidden) CRFs using neural features for ASR are underappreciated. This paper advances this approach, with clarified formulation, new development of CTC-CRFs and strong empirical results. Segmental CRFs [18] provide another solution to the alignment problem.

Model	State topology	Training objective	Locally/globally normalized
Regular HMM	HMM	$p(\boldsymbol{x}\|\boldsymbol{l})$	local
Regular CTC	CTC	$p(\boldsymbol{l}\|\boldsymbol{x})$	local
SS-LF-MMI	HMM	$p(\boldsymbol{l}\|\boldsymbol{x})$	local
CTC-CRF	CTC	$p(\boldsymbol{l}\|\boldsymbol{x})$	global
Seq2Seq	-	$p(\boldsymbol{l}\|\boldsymbol{x})$	local

Table 1. Comparison of different models for ASR. HMM topology denotes that labels (including silence) are modeled by multiple states with left-to-right transitions, possible self-loops and skips. CTC topology denotes the special state transitions used in CTC (including blank). Locally/globally normalized denotes the formulation of the model distribution. In defining the joint distribution of a model, local normalized models use conditional probability functions, while global normalized models use local un-normalized potential functions. A clarification on placing "local" for SS-LF-MMI is that SS-LF-MMI is cast as MMI-based discriminative training of a pseudo HMM, and the HMM model is local normalized. Seq2Seq does not use states to align label sequence $\boldsymbol{l}$ and observation sequence $\boldsymbol{x}$.

3. METHOD

3.1. Model definition

For our CRF model, the conditional probability of the hidden state sequence $\boldsymbol{\pi}$ given the observation sequence $\boldsymbol{x}$ is defined as:

$$p(\boldsymbol{\pi}|\boldsymbol{x};\boldsymbol{\theta}) = \frac{\exp(\phi(\boldsymbol{\pi},\boldsymbol{x};\boldsymbol{\theta}))}{\sum_{\boldsymbol{\pi}'}\exp(\phi(\boldsymbol{\pi}',\boldsymbol{x};\boldsymbol{\theta}))}$$

where $\boldsymbol{\pi}$ and $\boldsymbol{x}$ are of the same lengths (i.e. aligned). $\boldsymbol{\theta}$ is the model parameter. $\boldsymbol{\pi}$ is connected with $\boldsymbol{l}$ by a mapping $\mathcal{B}: S_\pi^T \rightarrow S_l^L$, which maps a state sequence $\boldsymbol{\pi}$ to a unique label sequence $\boldsymbol{l}$. S_π and S_l are the symbol tables for $\boldsymbol{\pi}$ and $\boldsymbol{l}$ respectively. T and L are the lengths of $\boldsymbol{\pi}$ and $\boldsymbol{l}$ respectively. Then $p(\boldsymbol{l}|\boldsymbol{x};\boldsymbol{\theta})$ can be defined as:

$$p(\boldsymbol{l}|\boldsymbol{x};\boldsymbol{\theta}) = \sum_{\boldsymbol{\pi}} p(\boldsymbol{\pi},\boldsymbol{l}|\boldsymbol{x};\boldsymbol{\theta}) = \sum_{\boldsymbol{\pi}\in\mathcal{B}^{-1}(\boldsymbol{l})} p(\boldsymbol{\pi}|\boldsymbol{x};\boldsymbol{\theta})$$

The CML objective is defined as $\mathcal{J}_{CRF}(\boldsymbol{\theta}) = \log p(\boldsymbol{l}|\boldsymbol{x};\boldsymbol{\theta})$, and the gradient of the objective is

$$\frac{\mathcal{J}_{CRF}(\boldsymbol{\theta})}{\partial\boldsymbol{\theta}} = \mathbb{E}_{p(\boldsymbol{\pi}|\boldsymbol{l},\boldsymbol{x})}\left[\frac{\partial\phi(\boldsymbol{\pi},\boldsymbol{x};\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}\right] - \mathbb{E}_{p(\boldsymbol{\pi}'|\boldsymbol{x})}\left[\frac{\partial\phi(\boldsymbol{\pi}',\boldsymbol{x};\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}\right].$$

As commonly found in estimating CRFs, the gradient is the difference between empirical expectation and model expectation. The two expectations are similar to the calculations using the numerator graph and denominator graph in LF-MMI respectively [19]. Next we define the mapping $\mathcal{B}$ and the potential function $\phi(\boldsymbol{\pi},\boldsymbol{x};\boldsymbol{\theta})$. For simplicity, we omit $\boldsymbol{\theta}$ in the following sections.

3.2. State topology

As discussed in Section 3.1, the only restriction of $\mathcal{B}$ is to map $\boldsymbol{\pi}$ to a unique $\boldsymbol{l}$. We choose to use the mapping represented by the CTC topology. The symbol table S_π contains the meaningful symbols (characters or phones) plus the blank symbol, and S_l only contains the meaningful symbols. We obtain $\boldsymbol{l}$ from $\boldsymbol{\pi}$ by first removing all repetitive symbols bewteen the blank symbols, and then removing all blank symbols, e.g. $\mathcal{B}(A---BB-B--A) = ABBA$.

It can be seen that any regular HMM topology with more than one state can also represent a mapping satisfying the above uniqueness restriction on $\mathcal{B}$. "Regular HMM" stands for the composite

HMM simply concatenating smaller HMMs. The CTC topology can be represented by HMM with a special topology. Unless otherwise stated, "HMM" in the following sections stands for regular HMM.

We choose the mapping represented by the CTC topology for two main reasons. First, CTC mapping has the smallest number of units in S_π among all possible mappings, by adding only one blank symbol into S_l. If we use the mapping represented by regular HMMs, the number of units in S_π is at least twice the number of units in S_l (note that the mapping represented by a single-state HMM can not map $\boldsymbol{\pi}$ to a unique $\boldsymbol{l}$). The second reason is that we prefer not to use the silence symbol, which will cause ad-hoc silence insertions in estimating the denominator LM. The blank symbol can absorb silences.

3.3. Potential function

The potential function of our CRF is defined as:

$$\phi(\boldsymbol{\pi}, \boldsymbol{x}) = \sum_{t=1}^{T} \log p(\pi_t|\boldsymbol{x}) + \log p(\boldsymbol{l})$$

where $\boldsymbol{l}$ is the corresponding label sequence $\boldsymbol{l} = \mathcal{B}(\boldsymbol{\pi})$. The first term $\sum_{t=1}^{T} \log p(\pi_t|\boldsymbol{x})$ is often referred to as the node potential. The second term $\log p(\boldsymbol{l})$ is the edge potential. $p(\boldsymbol{l})$ is defined based on the n-gram denominator LM of labels, like in LF-MMI [19]. Specifically, $p(\boldsymbol{l})$ is calculated as the path weight in a denominator WFST. The denominator WFST is the composition of the WFST representing the CTC topology and the n-gram denominator LM.

If we omit the edge potential $\log p(\boldsymbol{l})$, the potential function becomes self-normalized ($\sum_{\boldsymbol{\pi}'} \exp(\phi(\boldsymbol{\pi}', \boldsymbol{x})) = 1$) and the CRF model degrades to the regular CTC model.

3.4. Training and decoding

In training, we use the regular CTC objective as an auxiliary objective to help the convergence. The final objective function is

$$\mathcal{J}_{CTC-CRF} + \alpha \mathcal{J}_{CTC}.$$

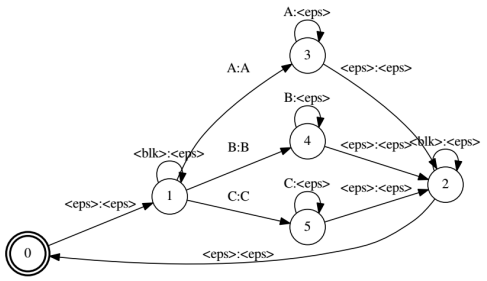
α is set to 0.1 in our experiments except WSJ, where α is set to 0.01.

In decoding, an external word-level LM is incorpolated. we search the decoding graph with the score function as:

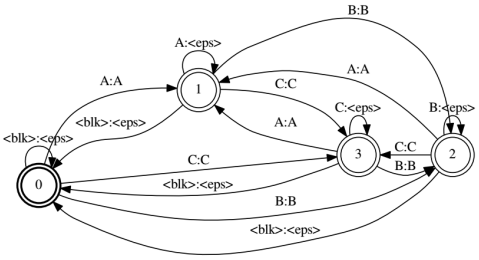
$$\log p(\boldsymbol{l}|\boldsymbol{x}) + \beta \log p_{LM}(\boldsymbol{l})$$

where $p_{LM}(\boldsymbol{l})$ is word-level LM probability of $\boldsymbol{l}$. β is set to 1.0 in our experiments. We use WFST based decoding similar to Eesen [2].

In both training and decoding, we need to construct a WFST representation of the CTC topology (called T.fst in Eesen). Eesen T.fst is not correct. An example of Eesen T.fst is shown in Fig 1(a). In this example, the symbol table S_l contains "A", "B" and "C", and S_π = $S_l \cup$ <blk>. "A:A" on the arc means consuming an input symbol "A" and outputting a symbol "A". "<eps>" means "no consuming" (no output). We can see that the Eesen T.fst makes mistakes when two adjacent identical labels appear in $\boldsymbol{l}$. The total number of corresponding paths in $\mathcal{B}^{-1}(\boldsymbol{l})$ will be mistakenly larger. We give a corrected construction of T.fst in Fig 1(b). State 1, state 2 and state 3 are considered to be the states corresponding to "A", "B" and "C". When encountering a blank symbol at these states, we come back to state 0 to consume blank symbols. State 1, state 2 and state 3 can jump to each other, consuming and outputting corresponding symbols.



(a) T.fst in Eesen



(b) Corrected T.fst representing CTC topology

Fig. 1. WFST representation of the CTC topology

4. EXPERIMENTS

Evaluation experiments are conducted on 80-h WSJ, 300-h Switchboard and 1000-h Librispeech datasets. The feature is the 40-dimentional fbank feature with delta and delta-delta features (120 dimentions in total). We use cepstral mean and variance normalization (CMVN) and subsampling of factor 3 to process the feature. Our acoustic model is a 6-layer bidirectional LSTM with 320 hidden units. Dropout is set to be 0.5 between every LSTM layer. The total number of parameters is 13M, much smaller than most end-to-end models. We use Pytorch to train CTC-CRFs. The optimizer is Adam with an initial learning rate of 0.001. When the cross-validation loss does not decrease, we decrease the learning rate to 0.0001. CTC-CRFs are trained without any pre-training, so we call it single-stage acoustic modeling.

The gradient computation in CTC-CRFs is implemented on GPUs. Unlike SS-LF-MMI, we do not insert any silence symbol when estimating the 4-gram denominator LM for the denominator forward-backward calculation. Moreover, SS-LF-MMI modifies the length of each utterance to one of 30 distinct lengths. In contrast, our implementation supports variable length sequences.

4.1. WSJ

The WSJ dataset contains 80-hour speech data. We randomly split the total 80 hours, using 95% data as the training data and using the other 5% data as the validation data. The evaluation dataset contains the harder dev93 set and the simpler eval92 set.

The results are shown in Table 2. The "bi-phone" used by SS-LF-MMI is the full bi-phone considering all possible context dependency. "SP" means speed perturbation for 3-fold data augmentation, marked with "Y" (using SP) and "N" (not using SP). The "gram-char" means using one or multiple characters as units (e.g. using "AB" and "A" as units). Unless otherwise stated, all language models are word based.

Model	Unit	LM	SP	dev93	eval92
SS-LF-MMI [4]	mono-phone	4-gram	Y	6.3%	3.1%
SS-LF-MMI [4]	bi-phone	4-gram	Y	6.0%	3.0%
SS-LF-MMI [3]	mono-char	3-gram	Y	-	5.2%
SS-LF-MMI [3]	bi-char	3-gram	Y	-	4.1%
SS-LF-MMI [3][1]	mono-char	3-gram	Y	-	5.4%
CTC [20]	mono-char	3-gram	N	9.21%	5.53%
Eesen [2]	mono-phone	3-gram	N	-	7.34%
Gram-CTC [21]	gram-char	-	N	-	6.75%
ESPNET [22]	mono-char	LSTM [2]	N	12.4%	8.9%
CTC	mono-phone	4-gram	N	10.81%	7.02%
CTC-CRF	mono-phone	4-gram	N	6.24%	3.90%
CTC-CRF	mono-phone	4-gram	Y	6.23%	3.79%
CTC	mono-char	4-gram	N	12.57%	8.59%
CTC-CRF	mono-char	4-gram	N	8.62%	5.19%
CTC-CRF	mono-char	4-gram	Y	8.22%	5.32%

[1] Using similar CTC topology
[2] Character based LSTM language model

Table 2. WSJ results

Model	Unit	LM	SP	SW	CH
SS-LF-MMI [4]	mono-phone	4-gram	Y	11.0%	20.7%
SS-LF-MMI [4]	bi-phone	4-gram	Y	9.8%	19.3%
SS-LF-MMI [3]	mono-char	4-gram	Y	13.3%	-
SS-LF-MMI [3]	bi-char	4-gram	Y	10.9%	20.6%
SS-LF-MMI [3][1]	mono-char	4-gram	Y	14.5%	-
Seq2Seq [6]	subword	LSTM[2]	N	11.8%	25.7%
CTC [23]	mono-char	n-gram	N	15.1%	26.3%
CTC [24]	subword	LSTM[2]	N	14.7%	26.2%
CTC [25]	word/mono-char	No LM	N	14.4%	24.0%
CTC	mono-phone	4-gram	N	12.9%	23.6%
CTC-CRF	mono-phone	4-gram	N	11.0%	21.0%
CTC-CRF	mono-phone	4-gram	Y	10.3%	19.7%
CTC	mono-char	4-gram	N	15.3%	26.7%
CTC-CRF	mono-char	4-gram	N	12.7%	24.0%
CTC-CRF	mono-char	4-gram	Y	11.4%	21.7%

[1] Using similar CTC topology
[2] Subword LSTM language model

Table 3. Switchboard results

Mono-phone/mono-char based CTC-CRFs achieve WERs of 3.79%/5.19% on the eval92 dataset respectively. These results show significant improvements over regular CTC, with 45.6%/39.6% relative WER reductions on eval92. Speed perturbation did not help much in the WSJ experiments. Compared to other systems, CTC-CRFs are only weaker than the bi-phone/bi-char based SS-LF-MMI models.

4.2. Switchboard

For the Swichboard dataset, we use the first 4000 utterances (total 5 hours) as the validation data. After removing some repetitive utterances, 286-hour data is used as the training data. The Eval2000 data, which contains both Swichboard evaluation dataset (SW) and Callhome evaluation dataset (CH), is used for evaluation.

The results are shown in Table 3. Compared to regular CTC, our mono-phone/mono-char based CTC-CRF systems achieve 14.7%/17.0% relative WER reductions. Speed perturbation gives 6.4%/10.2% relative improvements. CTC-CRFs obtain WERs of 10.3%/11.4% on the Switchboard evaluation set (SW). These results are significantly better than mono-phone/mono-char based SS-LF-MMI models, though slightly worse (nearly relative 5%) than their full bi-phone/bi-char based models. Comparing to other recent end-to-end systems, our CTC-CRF systems are consistently better.

Model	Unit	LM	SP	dev		test	
				clean	other	clean	other
LF-MMI [19]	tri-phone	4-gram	Y	-	-	4.28%	-
Seq2Seq [6]	subword	4-gram[1]	N	4.79%	14.31%	4.82%	15.30%
CTC [20]	mono-char	4-gram	N	5.10%	14.26%	5.42%	14.70%
CTC [26]	mono-char	4-gram	N	-	-	4.8%	14.5%
CTC	mono-phone	4-gram	N	4.64%	13.23%	5.06%	13.68%
CTC-CRF	mono-phone	4-gram	N	3.87%	10.28%	4.09%	10.65%
CTC	mono-char	4-gram	N	5.00%	14.51%	5.29%	15.26%
CTC-CRF	mono-char	4-gram	N	4.26%	12.11%	4.67%	12.49%

[1] Subword n-gram language model

Table 4. Librispeech results

WFST	dev		test	
	clean	other	clean	other
Eesen T.fst	3.90%	10.32%	4.11%	10.68%
Corrected T.fst	3.87%	10.28%	4.09%	10.65%

Table 5. WERs with different T.fst on Librispeech test-clean set

WFST	TLG size	decoding time
Eesen T.fst	208M	700s
Corrected T.fst	181M	672s

Table 6. The decoding graph size and the time (excluding the neural network computation) in decoding Librispeech test-clean set. The language model is the official tri-gram language model.

4.3. Librispeech

We use 95% of the total 960h data as the training data, the other 5% as the validation data. The official dev-clean, dev-other, test-clean and test-other sets are used for evaluation.

Speed perturbation is not used in the Librispeech experiments due to time and computational limitation. Mono-phone/mono-char based CTC-CRF systems obtain 19.1%/11.7% relative WER reductions over regular CTC. Mono-char based CTC-CRF performs better than other character/subword based systems, and mono-phone based CTC-CRF outperforms the regular LF-MMI system [19] (4.28% on the test-clean set, with speed perturbation and i-vector).

The training speed on Librispeech is approximately 1.3 hours per epoch for the mono-phone CTC-CRF system, with 4 Tesla-P100 GPUs. The model converges after 35 epochs.

4.4. Analysis about T.fst and alignments

T.fst is used in both training and decoding. In training, we must use the corrected T.fst, otherwise $p(\boldsymbol{l}|\boldsymbol{x}) = \sum_{\boldsymbol{\pi}\in\mathcal{B}^{-1}(\boldsymbol{l})} p(\boldsymbol{\pi}|\boldsymbol{x})$ is not a valid probability distribution. In decoding, we also experiment with Eesen's T.fst. The WERs obtained with different T.fst in decoding are shown in Table 5. Decoding with the corrected T.fst performs slightly better. Moreover, as shown in Table 6, when using the corrected T.fst, the decoding graph (TLG) size is smaller and the decoding speed is faster.

In our experiments, we find that the alignments from CTC-CRFs are similar to those from CTC, with peaks of symbols. Some useful techniques for CTC decoding [14] could be applied for CTC-CRFs.

5. CONCLUSIONS

We propose a framework for single-stage acoustic modeling based on CRFs with CTC topology. CTC-CRFs achieve competitive results on WSJ, Switchboard and Librispeech datasets. Future work includes using other units with larger context (e.g. bi-phones, bi-chars, subwords), and exploring new potential function ϕ.

6. REFERENCES

[1] George E Dahl, Dong Yu, Li Deng, and Alex Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on audio, speech, and language processing*, vol. 20, no. 1, pp. 30–42, 2012.

[2] Yajie Miao, Mohammad Gowayyed, and Florian Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *Proc. ASRU*, 2015, pp. 167–174.

[3] Hossein Hadian, Hossein Sameti, Daniel Povey, and Sanjeev Khudanpur, "End-to-end speech recognition using lattice-free MMI," in *Proc. Interspeech*, 2018, pp. 12–16.

[4] Hossein Hadian, Hossein Sameti, Daniel Povey, and Sanjeev Khudanpur, "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.

[5] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. ICML*, 2014, pp. II–1764–II–1772.

[6] Albert Zeyer, Kazuki Irie, Ralf Schlter, and Hermann Ney, "Improved training of end-to-end attention models for speech recognition," in *Proc. Interspeech*, 2018, pp. 7–11.

[7] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proc. ICML*, 2006, pp. 369–376.

[8] Chung-Cheng Chiu, Tara N Sainath, Yonghui Wu, Rohit Prabhavalkar, Patrick Nguyen, Zhifeng Chen, Anjuli Kannan, Ron J Weiss, Kanishka Rao, Ekaterina Gonina, et al., "State-of-the-art speech recognition with sequence-to-sequence models," in *Proc. ICASSP*, 2018, pp. 4774–4778.

[9] Alex Graves, "Sequence transduction with recurrent neural networks," *arXiv preprint arXiv:1211.3711*, 2012.

[10] Eric Battenberg, Jitong Chen, Rewon Child, Adam Coates, Yashesh Gaur, Yi Li, Hairong Liu, Sanjeev Satheesh, Anuroop Sriram, and Zhenyao Zhu, "Exploring neural transducers for end-to-end speech recognition," in *Proc. ASRU*, 2017, pp. 206–213.

[11] John Lafferty, Andrew McCallum, and Fernando Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. ICML*, 2001.

[12] Albert Zeyer, Eugen Beck, Ralf Schlüter, and Hermann Ney, "CTC in the context of generalized full-sum HMM training," in *Proc. Interspeech*, 2017, pp. 944–948.

[13] Georg Heigold, Hermann Ney, Patrick Lehnen, Tobias Gass, and Ralf Schlüter, "Equivalence of generative and log-linear models," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 1138–1148, 2011.

[14] Zhehuai Chen, Yimeng Zhuang, Yanmin Qian, and Kai Yu, "Phone synchronous speech recognition with CTC lattices," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 25, no. 1, pp. 90–101, 2017.

[15] Brian Kingsbury, "Lattice-based optimization of sequence classification criteria for neural-network acoustic modeling," in *Proc. ICASSP*, 2009, pp. 3761–3764.

[16] Kanishka Rao, Andrew Senior, and Haşim Sak, "Flat start training of CD-CTC-sMBR LSTM RNN acoustic models," in *Proc. ICASSP*, 2016, pp. 5405–5409.

[17] Asela Gunawardana, Milind Mahajan, Alex Acero, and John C Platt, "Hidden conditional random fields for phone classification," in *Ninth European Conference on Speech Communication and Technology*, 2005.

[18] Liang Lu, Lingpeng Kong, Chris Dyer, Noah A. Smith, and Steve Renals, "Segmental recurrent neural networks for end-to-end speech recognition," in *Proc. Interspeech*, 2016, pp. 385–389.

[19] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Proc. Interspeech*, 2016, pp. 2751–2755.

[20] Yingbo Zhou, Caiming Xiong, and Richard Socher, "Improving end-to-end speech recognition with policy learning," in *Proc. ICASSP*, 2018, pp. 5819–5823.

[21] Hairong Liu, Zhenyao Zhu, Xiangang Li, and Sanjeev Satheesh, "Gram-CTC: Automatic unit selection and target decomposition for sequence labelling," in *Proc. ICML*, 2017, pp. 2188–2197.

[22] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proc. Interspeech*, 2018, pp. 2207–2211.

[23] Geoffrey Zweig, Chengzhu Yu, Jasha Droppo, and Andreas Stolcke, "Advances in all-neural speech recognition," in *Proc. Interspeech*, 2017, pp. 4805–4809.

[24] Thomas Zenkel, Ramon Sanabria, Florian Metze, and Alex Waibel, "Subword and crossword units for CTC acoustic models," in *Proc. Interspeech*, 2018, pp. 396–400.

[25] Kartik Audhkhasi, Brian Kingsbury, Bhuvana Ramabhadran, George Saon, and Michael Picheny, "Building competitive direct acoustics-to-word models for English conversational speech recognition," in *Proc. ICASSP*, 2018, pp. 4759–4763.

[26] V Liptchinsky, G Synnaeve, and R Collobert, "Letter-based speech recognition with gated convnets," *arXiv preprint arXiv:1712.09444*, 2017.