Phoneme based Neural Transducer for Large Vocabulary Speech Recognition

Wei Zhou, Simon Berger, Ralf Schlüter, Hermann Ney

Human Language Technology and Pattern Recognition, RWTH Aachen University, Aachen, Germany

Paper ID: 4200

Lehrstuhl Informatik 6
Human Language Technology and Pattern Recognition

Motivation

Classical hybrid hidden Markov model (HMM)

- pros: flexibility (modularity), scalability to low-resource tasks
- cons: complexity, inconsistency of modeling

End-to-end automatic speech recognition (ASR)

- pros: simplicity, consistent training & inference
- cons: flexibility, scalability, amount of data & training time

Goal: join the advantages of both approaches

Phoneme-based Neural Transducer

Model definition

$$p(a_1^S \mid x_1^{T'}) = \sum_{(y,s)_1^U : a_1^S} p(y_1^U, s_1^U \mid h_1^T)$$

$x_1^{T'}$ - input feature sequence
h_1^T - encoder output $f^{\text{enc}}(x_1^{T'})$
a_1^S - output label sequence $(a \in V)$
y_1^U - alignment sequence $a_1^S \rightarrow h_1^T$
s_1^U - transition sequence $y_u \rightarrow a_{s_u}$
(in this work: $u = t, U = T$)

RNA alignment label topology [Sak+ 2017], [Tripathi+ 2019]

- y_1^U: each a_s occurs only once and blank label ϵ elsewhere
- s_1^U: fully defined by y_1^U as $s_u = s_{u-1} + (1 - \delta_{y_u,\epsilon})$

$$p(a_1^S \mid x_1^{T'}) = \sum_{(y,s)_1^U : a_1^S} \prod_{u=1}^{U} p(y_u \mid y_1^{u-1}, h_1^T) = \sum_{(y,s)_1^U : a_1^S} \prod_{u=1}^{U} p_\theta(y_u \mid a_{s_{u-1}-k+1}^{s_{u-1}}, h_1^T)$$

- context size k (default 1): local dependency (co-articulation)

HMM alignment label topology

- y_1^U: each a_s can loop for multiple steps and no blank ϵ

$$p(a_1^S \mid x_1^{T'}) = \sum_{(y,s)_1^U : a_1^S} \prod_{u=1}^{U} p(s_u \mid y_1^{u-1}, s_1^{u-1}, h_1^T) \cdot p(y_u \mid y_1^{u-1}, s_1^u, h_1^T)$$

$$p(s_u \mid y_1^{u-1}, s_1^{u-1}, h_1^T) = \begin{cases} q_\theta(y_u = y_{u-1} \mid a_{s_{u-1}-k}^{s_{u-1}-1}, h_1^T), & s_u = s_{u-1} \\ 1 - q_\theta(y_u = y_{u-1} \mid a_{s_{u-1}-k}^{s_{u-1}-1}, h_1^T), & s_u = s_{u-1} + 1 \end{cases}$$

$$p(y_u \mid y_1^{u-1}, s_1^u, h_1^T) = \begin{cases} \delta_{y_u, y_{u-1}}, & s_u = s_{u-1} \\ q_\theta(y_u \mid a_{s_u-k}^{s_u-1}, h_1^T), & s_u = s_{u-1} + 1 \end{cases}$$

Decision & decoding

- external word-level language model (LM) and lexicon
- no internal LM [Variani+ 2020] applied: suppressed negative effect

$$\begin{aligned} x_1^{T'} \rightarrow \tilde{w}_1^N &= \arg\max_{w_1^N} p^\lambda(w_1^N) \sum_{a_1^S : w_1^N} p(a_1^S \mid x_1^{T'}) \\ &= \arg\max_{w_1^N} p^\lambda(w_1^N) \sum_{(y,s)_1^U : a_1^S : w_1^N} p(y_1^U, s_1^U \mid h_1^T) \quad \text{full-sum} \\ &\approx \arg\max_{w_1^N} p^\lambda(w_1^N) \max_{(y,s)_1^U : a_1^S : w_1^N} p(y_1^U, s_1^U \mid h_1^T) \quad \text{Viterbi} \end{aligned}$$

Simplification and Extension

Simplified NN architecture

- recurrent neural network transducer (RNN-T) [Graves 2012]
- encoder: 6×512 bidirectional long short-term memory (BLSTM) with subsampling of factor 2 using max-pooling
- feed-forward neural network (FFNN)-based prediction network
- joint network (element-wise addition) and a final softmax
- footprint: about 30M parameters

Viterbi training

- full-sum (FS) over all alignments: time and memory consuming
- frame-wise cross-entropy (CE) loss w.r.t. $p(y_1^U, s_1^U \mid h_1^T)$ and a fixed external alignment
- enable more training techniques for speed and performance

Word boundary-based phoneme label augmentation

- end-of-word (EOW) phonemes: $2 \times |V|$
- start-of-word (SOW) + EOW phonemes: $4 \times |V|$

Experiments and Word Error Rate (WER) Results

Setup

- TED-LIUM Release 2 (TLv2)
- 300h Switchboard (SWBD): Hub5'00 (dev) and Hub5'01 (test)
- recognition: full-sum decoding with a 4-gram word-level LM

Label unit & topology

Phoneme Label	TLv2-dev		Hub5'00	
	RNA	HMM	RNA	HMM
original	7.6	9.3	14.0	15.4
EOW-augmented	**6.9**	8.8	**13.4**	14.5
+ SOW-augmented	7.3	9.0	13.5	14.8

- EOW-augmented phonemes + RNA topology: further experiments

Viterbi alignment & label position u_s

Alignment	u_s	TLv2-dev	Hub5'00
hybrid HMM	segBeg	7.2	13.7
	segMid	7.4	13.8
	segEnd	**6.9**	**13.4**
CTC		7.2	**13.4**

- u_s: positions in y_1^U where a_s occurs
- stable training procedure: various alignment properties

References

- [Sak+ 2017] Hasim Sak et al., "Recurrent Neural Aligner: An Encoder-Decoder Neural Network Model for Sequence to Sequence Mapping", Interspeech 2017
- [Tripathi+ 2019] Anshuman Tripathi et al., "Monotonic Recurrent Neural Network Transducer and Decoding Strategies", ASRU 2019
- [Variani+ 2020] Ehsan Variani et al., "Hybrid Autoregressive Transducer (HAT)", ICASSP 2020
- [Graves 2012] Alex Graves, "Sequence Transduction with Recurrent Neural Networks", 2012, https://arxiv.org/abs/1211.3711
- [Karita+ 2019] Shigeki Karita et al., "A Comparative Study on Transformer vs RNN in Speech Applications", ASRU 2019
- [Han+ 2017] Kyu J. Han et al., "The CAPIO 2017 Conversational Speech Recognition System", 2018, http://arxiv.org/abs/1801.00059
- [Zhou+ 2020] Wei Zhou et al., "The RWTH ASR system for TED-LIUM release 2: Improving Hybrid HMM with SpecAugment", ICASSP 2020
- [Raissi+ 2020] Tina Raissi et al., "Context-Dependent Acoustic Modeling without Explicit Phone Clustering", Interspeech 2020
- [Zoph+ 2019] Barret Zoph et al., "SpecAugment: A Simple Augmentation Method for Automatic Speech Recognition", Interspeech 2019
- [Tüske+ 2020] Zoltán Tüske et al., "Single Headed Attention based Sequence-to-sequence Model for State-of-the-Art Results on Switchboard", Interspeech 2020

Further WER Results

Context & efficiency

Train	Ch-unk	k	TLv2-dev		Hub5'00	
			WER	min/ep	WER	min/ep
Vit.	yes	1	**6.9**	93	**13.4**	132
		2	7.0	n.a.	13.6	n.a.
	no	1	7.2		14.1	
		2	7.0		13.8	
		∞	7.9		15.3	
FS			8.7	250	16.4	372

- better performance and efficiency compared to FS training

Ablation study

Training	TLv2 dev	Hub 5'00
default	6.9	13.4
- SpecAugment	8.5	14.6
- chunking	7.2	14.1
- encoder loss	7.3	14.0
- label smooth	8.0	14.2
- lossBoost$_{u_s}$	9.9	16.4
+ sampling	6.9	**12.9**

Overall WER on TLv2 and SWBD

- LSTM LM one pass + Transformer (Trafo) LM rescoring
- no seq-discriminative / speaker-adaptive training + less epochs
- TLv2: comparable to state-of-the-art (SOTA)

Work	Modeling			LM	TLv2	
	#Epoch	Approach	Label		dev	test
[Karita+ 2019]	100	Attention	subword	RNN	9.3	8.1
[Han+ 2017]	-	hybrid HMM	triphone		7.1	7.7
[Zhou+ 2020]	35			LSTM	5.6	6.0
				Trafo	5.1	5.6
this	50	Transducer	phoneme	LSTM	5.9	6.3
				Trafo	5.4	6.0

- SWBD: approaching SOTA

Work	Modeling			LM	Hub 5'00	Hub 5'01
	#Epoch	Approach	Label			
[Raissi+ 2020]	90	hybrid HMM	phoneme-state	LSTM	11.7	-
[Zoph+ 2019]	760	Attention	subword	RNN	10.5	-
[Tüske+ 2020]	250			LSTM	9.8	10.1
this	100	Transducer	phoneme	LSTM	11.5	11.5
				Trafo	11.2	11.2

Conclusion

A simple and competitive phoneme-based neural transducer approach

- advantages of both classical and end-to-end approaches
- utilize local dependency of phonemes: simplified NN with small footprint and straightforward LM integration
- stable and efficient training using frame-wise CE loss
- RNA topology: better than HMM topology for transducer modeling
- EOW-augmented phonemes: consistent improvement
- phonetic context size of one + chunk-wise Viterbi training: best performance

Acknowledgements

European Research Council
Established by the European Commission

This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No 694537, project "SEQCLAS") and from a Google Focused Award. 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.

Email: {zhou, schlueter, ney}@cs.rwth-aachen.de, simon.berger@rwth-aachen.de