



首届-东南亚非通用语种智能处理技术与应用研讨会

数据高效的多语言与跨语言语音识别

欧智坚

清华大学·语音处理与机器智能(SPMI)实验室

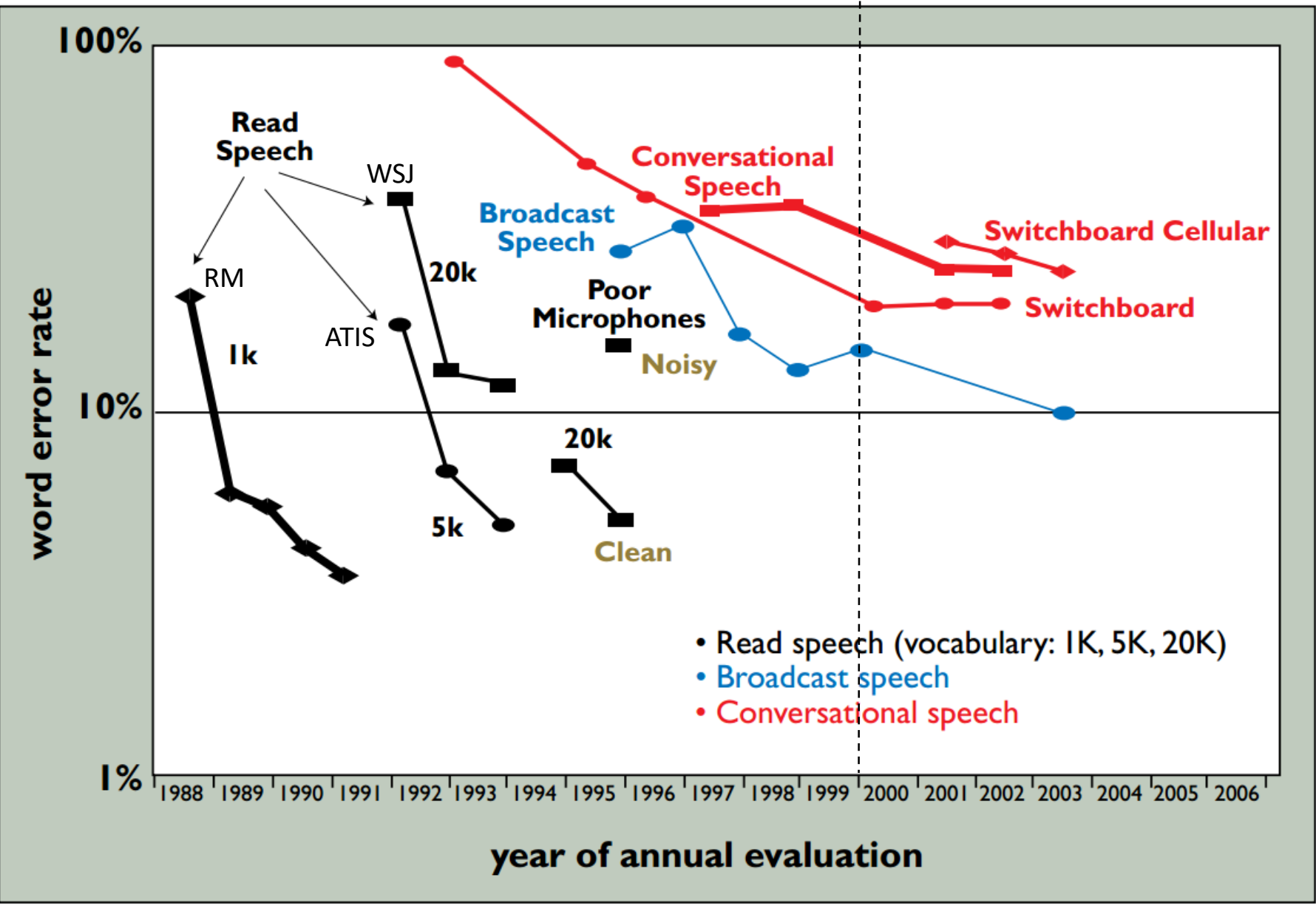
北京信息科学与技术国家研究中心

<http://oa.ee.tsinghua.edu.cn/ouzhijian/>

2021/12/18

提纲

- 一、语音识别简史与基础
- 二、端到端语音识别
- 三、数据高效
- 四、多语言与跨语言语音识别
- 五、总结及展望

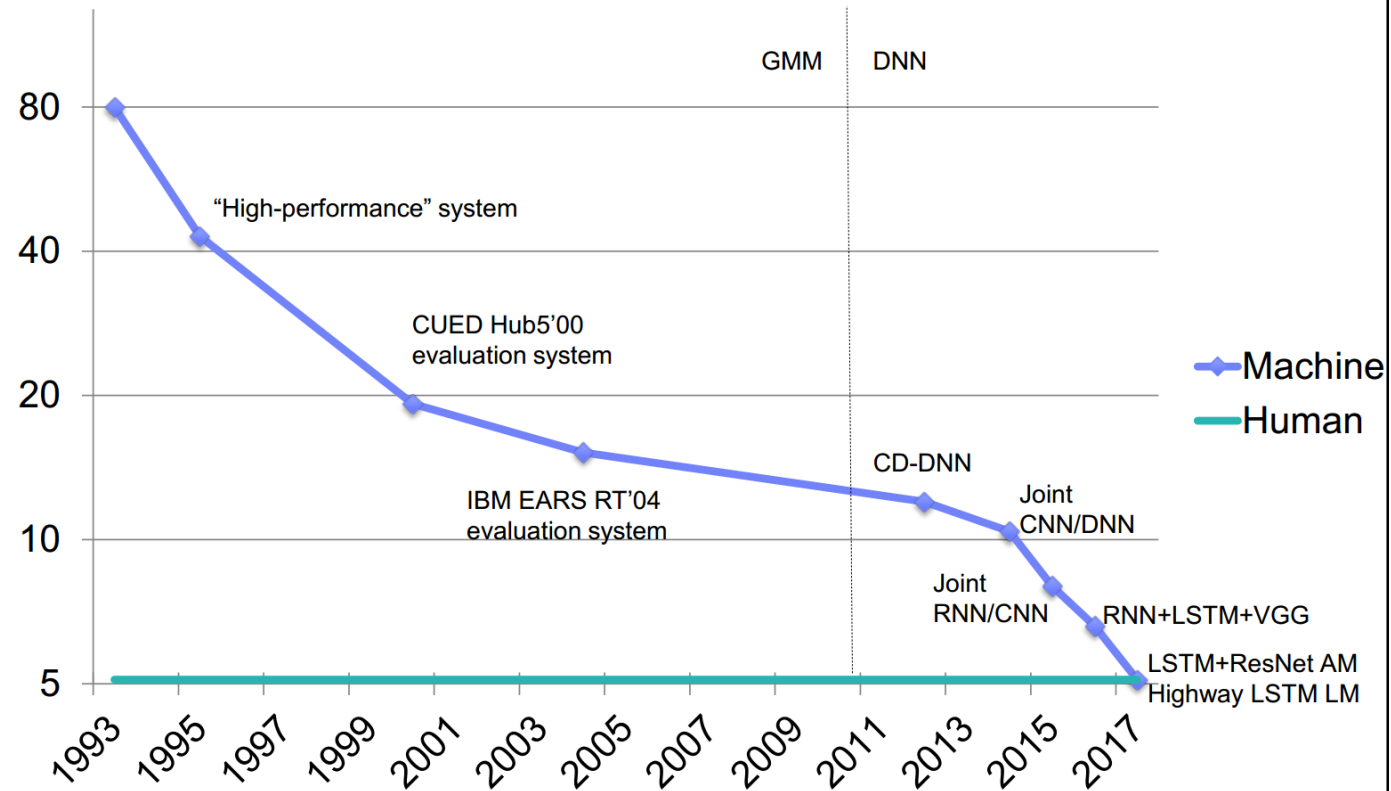
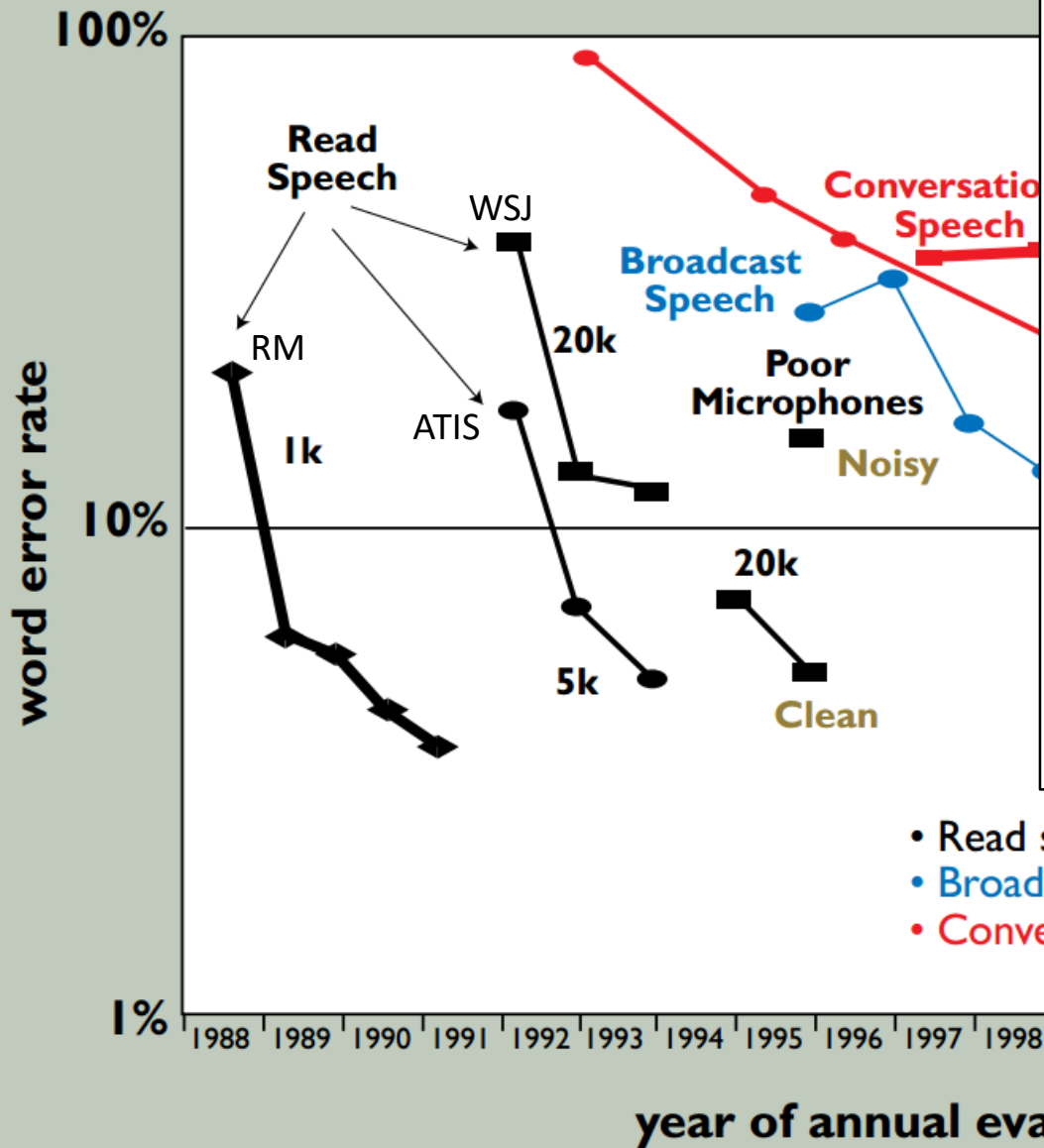


ASR brief history

1970 – 2010: 1st Generation

HMM	<ul style="list-style-type: none">• F. Jelinek, “Continuous speech recognition by statistical methods”, Proc. of the IEEE, 1976.• J. Baker, “The DRAGON system--An overview”, T-ASSP, 1975.
GMM	<ul style="list-style-type: none">• B.H. Juang, “Maximum-likelihood estimation for mixture multivariate stochastic observations of Markov chains”, AT&T Technical Journal, 1985.
N-gram, Smoothing	<ul style="list-style-type: none">• F. Jelinek & R.L. Mercer, “Interpolated estimation of Markov source parameters from sparse data”, Proc. Workshop on Pattern Recognition in Practice, 1980.• F. Jelinek, “The development of an Experimental Discrete Dictation Recognizer”, Proc. of the IEEE, 1985.
Tree based state tying	<ul style="list-style-type: none">• S. Young, J.J. Odell, P.C. Woodland, “Tree-based state tying for high accuracy acoustic modeling”, HLT workshop, 1994.
MAP, MLLR	<ul style="list-style-type: none">• C.H. Lee, C.H. Lin, B.H. Juang, “A study on speaker adaptation of the parameters of continuous density hidden Markov models”, T-IP, 1991.• C.J. Leggetter & P.C. Woodland, “Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models”, Computer Speech and Language, 1995.
fMLLR, Speaker adaptive training	<ul style="list-style-type: none">• M.J.F. Gales, “Maximum likelihood linear transformations for HMM-based speech recognition”, Computer Speech and Language, 1998.
WFST	<ul style="list-style-type: none">• M. Mohri. Finite-State Transducers in Language and Speech Processing. Computational Linguistics, 1997.• M. Mohri, F. Pereira, and M. Riley, “Speech Recognition with Weighted Finite-State Transducers”, 2008.
Discriminative Training, MMI, MPE	<ul style="list-style-type: none">• D. Povey, “Discriminative training for large vocabulary speech recognition”, Ph.D. dissertation, 2003.

Progress on Switchboard (Hub5'00 SWB testset*)



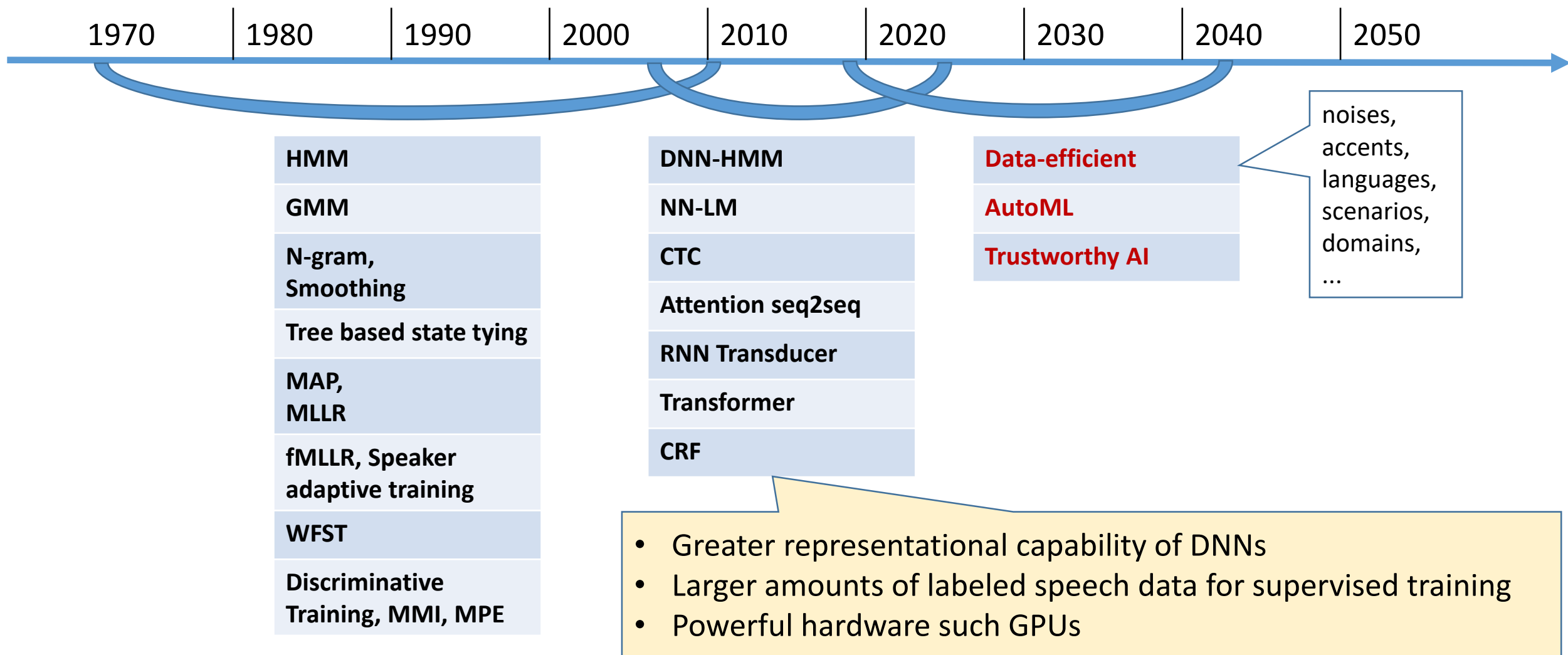
Man vs. machine in conversational speech recognition, George Saon, ASRU2017 Invited talk.

ASR brief history

2011 – now: 2nd Generation

DNN-HMM	<ul style="list-style-type: none">• A. Mohamed, G. Dahl, and G. Hinton, “Deep belief networks for phone recognition”, NIPS Workshop Deep Learning for Speech Recognition and Related Applications, 2009.• G. Dahl, et al, “Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition”, T-ASLP, 2012.• F. Seide, G. Li, and D. Yu, “Conversational speech transcription using context-dependent deep neural networks”, Interspeech, 2011.• D. Povey, et al, "Purely sequence-trained neural networks for ASR based on lattice-free MMI", Interspeech 2016.
NN-LM	<ul style="list-style-type: none">• Bengio, et al, “A Neural Probabilistic Language Model”, NIPS, 2001.• Mikolov, et al, "Recurrent neural network based language model", Interspeech, 2010.
CTC	<ul style="list-style-type: none">• A. Graves, et al, “Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks”, ICML, 2006.• H. Sak, et al, “Learning acoustic frame labeling for speech recognition with recurrent networks”, ICASSP, 2015.• Y. Miao, et al, “EESSEN: End-to-end speech recognition using deep RNN models and WFST-based decoding”, ASRU, 2015.
Attention seq2seq	<ul style="list-style-type: none">• D. Bahdanau, et al, “Neural machine translation by jointly learning to align and translate”, ICLR 2015.• J. K. Chorowski, et al, “Attention-based models for speech recognition,” NIPS, 2015.• W. Chan, et al @ google, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition”, ICASSP, 2016.
RNN Transducer	<ul style="list-style-type: none">• A. Graves, “Sequence transduction with recurrent neural networks,” ICML 2012 Workshop on Representation Learning.• E. Battenberg, et al @ Baidu, “Exploring neural transducers for end-to-end speech recognition”, ASRU 2017.• K. Rao, et al @ Google, “Exploring architectures, data and units for streaming end-to-end speech recognition with RNN-transducer”, ASRU 2017
Transformer	<ul style="list-style-type: none">• A. Vaswani, et al @ google, "Attention Is All You Need", NIPS, 2017.
CRF	<ul style="list-style-type: none">• H. Xiang, Z. Ou. "CRF-based Single-stage Acoustic Modeling with CTC Topology", ICASSP, 2019.

New-generation ASR

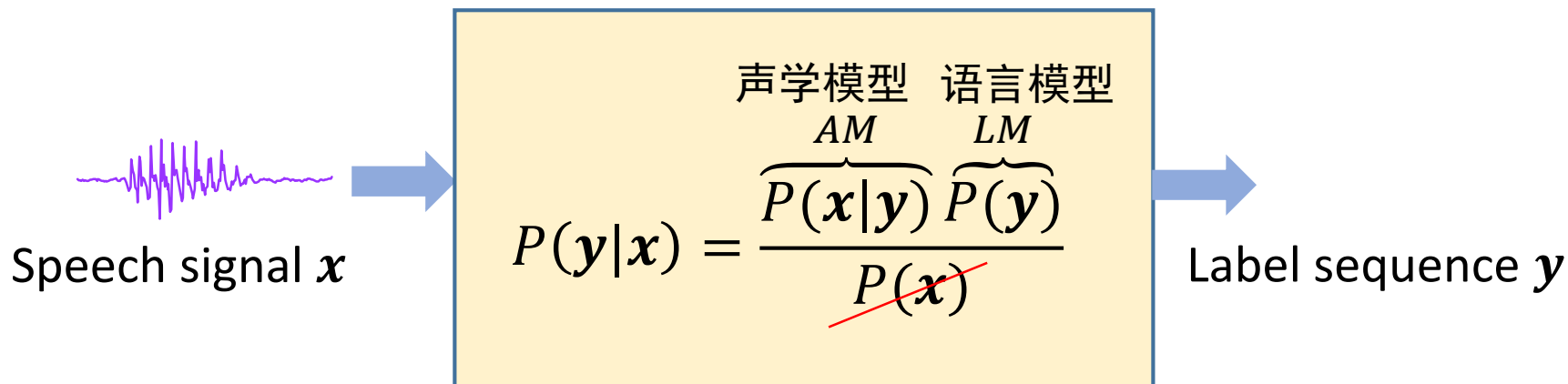


ASR: Basics

ASR (Automatic Speech Recognition) is a seq. discriminative problem

- For acoustic observations $\mathbf{x} \triangleq x_1, \dots, x_T$, find the most likely labels $\mathbf{y} \triangleq y_1, \dots, y_L$

- How to obtain $p(\mathbf{y} | \mathbf{x})$
- How to handle alignment, since $L \neq T$



Labels

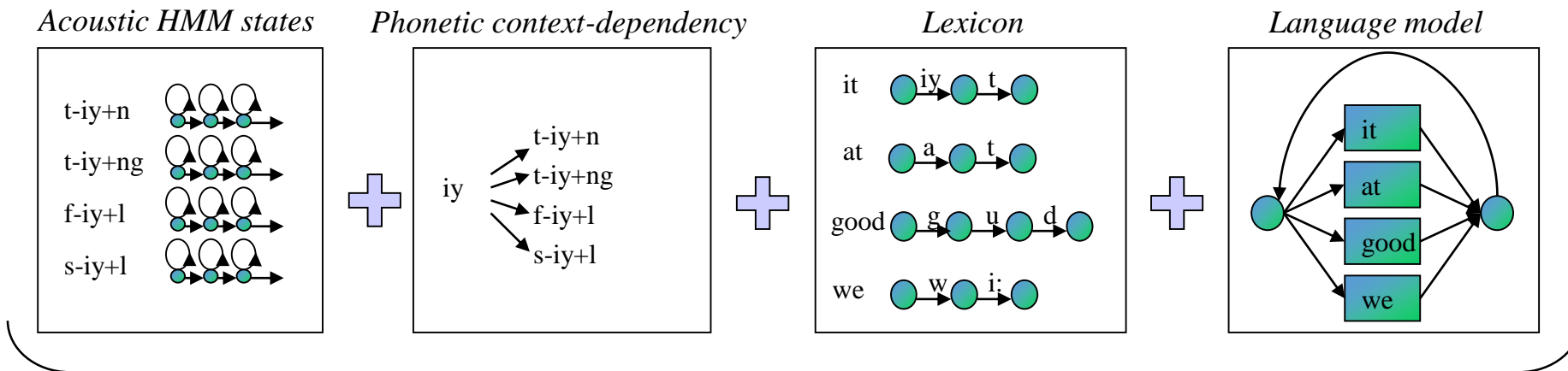
$L \neq T$

y						π_7	π_8
y_1					π_6		
\vdots							
y_L	π_1	π_2					

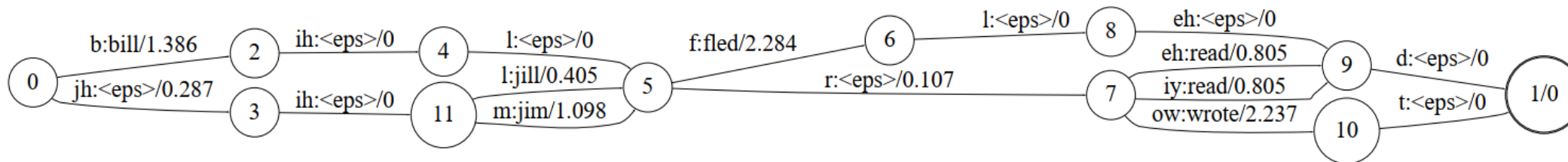
Observations $\mathbf{x} = x_1 \dots x_T$

Example of alignment

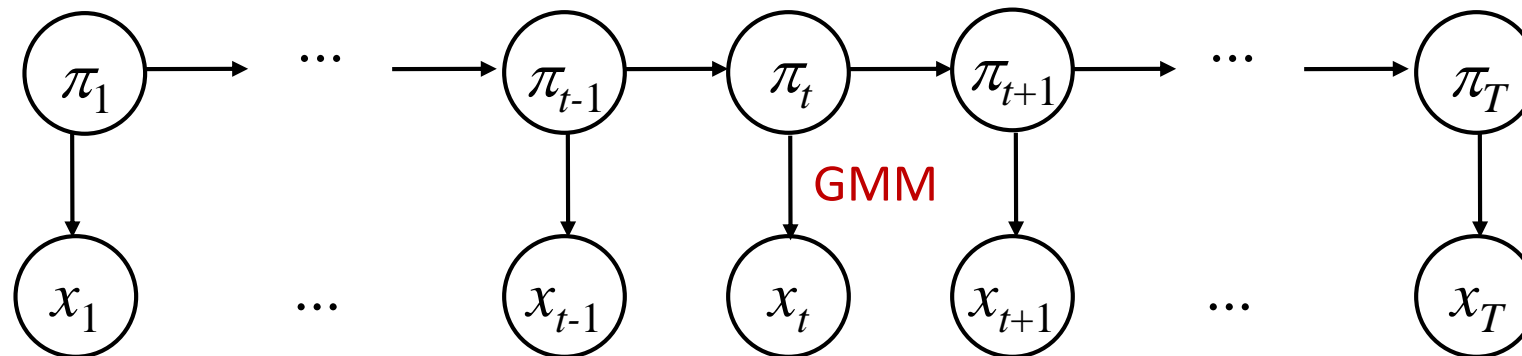
GMM-HMM: state transitions



State transitions in π are determined by a state transition graph (WFST), constrained by \uparrow



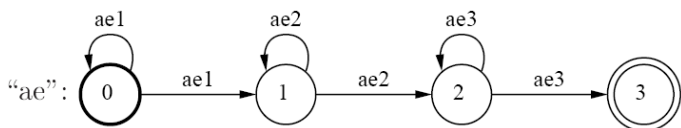
A path $\pi \triangleq \pi_1, \dots, \pi_T$ uniquely determines a label sequence y , but not vice versa.



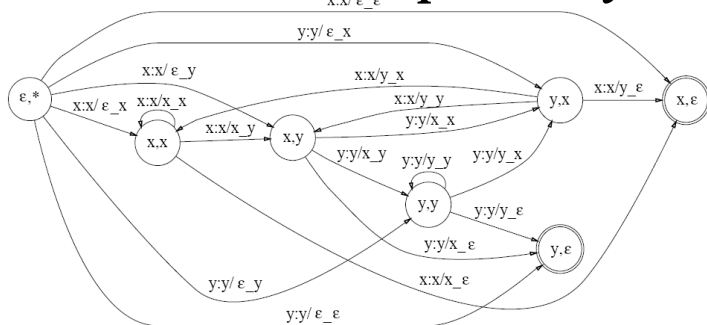
WFST

- WFSTs (weighted finite-state transducers) for Viterbi decoding
 - Pioneered by AT&T in late 1990's [Mohri et al., 2008]

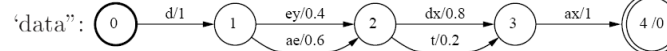
Acoustic HMMs: H



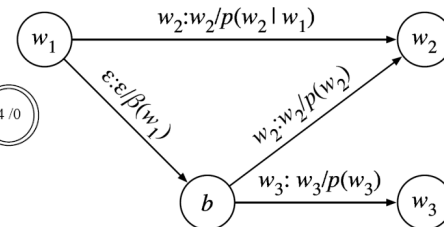
Phonetic context-dependency: C



Lexicon: L



Language model: G



Composed and optimized into a single WFST

$$N = \min \left(\det \left(H \circ \det \left(C \circ \det \left(L \circ G \right) \right) \right) \right)$$

which represents $p(\pi_{t+1} | \pi_t)$ and is used in Viterbi decoder.

Well implemented in Kaldi toolkit <https://github.com/kaldi-asr/kaldi>

DNN-HMM

- ASR state-of-the-art: DNNs of various network architectures (MLP, LSTM, CNN, Transformer, etc.), initially DNN-HMM

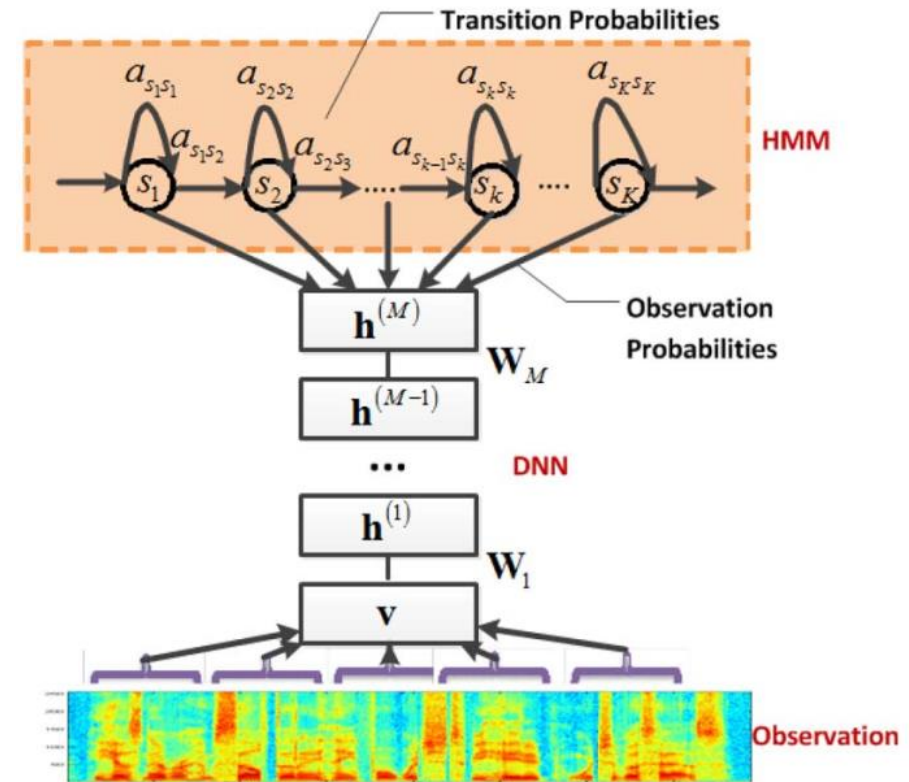
State posterior prob.
estimated from the DNN, which needs
frame-level alignments

Can be ignored.

$$p(x_t|\pi_t) = \frac{p(\pi_t|x_t)p(x_t)}{p(\pi_t)}$$

State prior prob.
estimated from the training data

- Conventionally, multi-stage
 - monophone GMM-HMM
 - alignment & triphone tree building
 - triphone GMM-HMM
 - alignment
 - triphone DNN-HMM



[Dahl, et al., TASLP 2012]

G. Dahl, et al., "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition", TASLP, 2012.

提纲

- 一、语音识别简史与基础
- 二、端到端语音识别
- 三、数据高效
- 四、多语言与跨语言语音识别
- 五、总结及展望

Advancing to end-to-end ASR: motivation

- End-to-end in the sense that:

- Eliminate the construction of GMM-HMMs and phonetic decision-trees, and can be trained from scratch (**flat-start** or **single-stage**)

- In a more strict/ambitious sense:

- Remove the need for a Pronunciation Lexicon (ProLex) and, even further, train the acoustic and language models jointly rather than separately
- Trained to optimize criteria that are related to the final evaluation metric that we are interested in (typically, word error rate)

- Motivation

- Simplify system pipeline, reduce expert knowledge and labor (such as compiling the ProLex, building phonetic decision trees)

Advancing to end-to-end ASR: techniques

ASR is a *sequence discriminative* problem

- For acoustic observations $\mathbf{x} \triangleq x_1, \dots, x_T$, find the most likely labels $\mathbf{y} \triangleq y_1, \dots, y_L$

- How to obtain $p(\mathbf{y} | \mathbf{x})$
- How to handle alignment, since $L \neq T$

- Need a differentiable sequence-level loss of mapping acoustic sequence \mathbf{y} to label sequence \mathbf{x}

- Explicitly:** introduce hidden state sequence $\boldsymbol{\pi}$, as in Connectionist Temporal Classification (CTC), RNN Transducer (RNNT), Conditional Random Field (CRF)
- Implicitly:** as in Attention based Encoder-Decoder (AED)

Labels

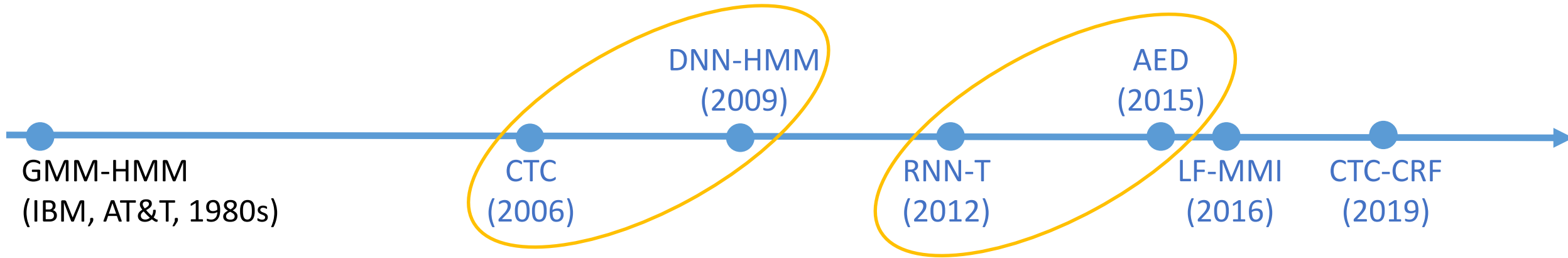
$L \neq T$

\mathbf{y}									
\parallel							π_7	π_8	
y_1						π_6			
\vdots			π_3	π_4	π_5				
y_L	π_1	π_2							

Observations $\mathbf{x} = x_1 \dots x_T$

Example of explicit alignment

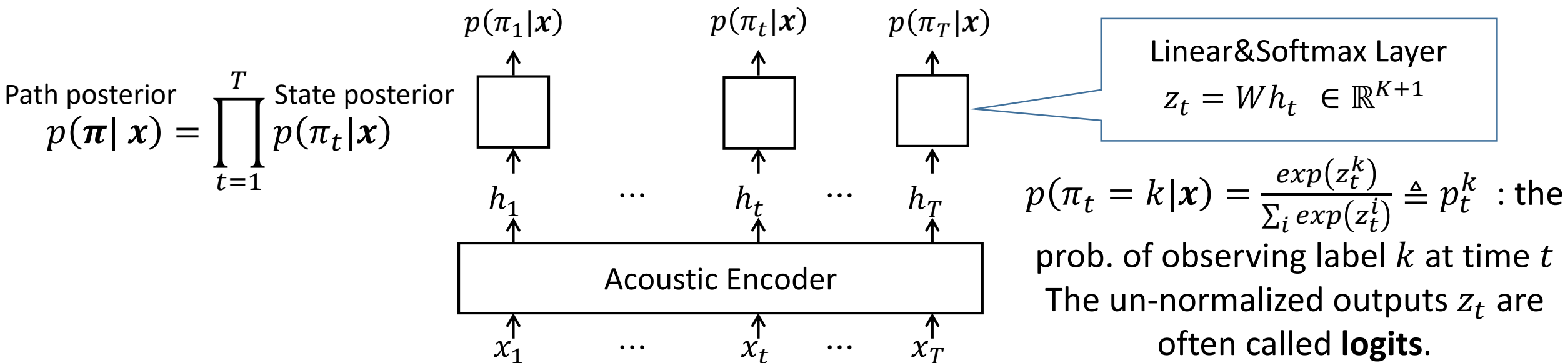
History



- [CTC] Graves, et al., “Connectionist Temporal Classification: Labelling unsegmented sequence data with RNNs”, ICML 2006.
- [DNN-HMM] A. Mohamed, et al., “Deep belief networks for phone recognition”, NIPS Workshop Deep Learning for Speech Recognition and Related Applications, 2009.
- [RNNT] A. Graves, “Sequence transduction with recurrent neural networks”, ICML 2012 Workshop on Representation Learning.
- [AED] D. Bahdanau, et al., “Neural machine translation by jointly learning to align and translate”, ICLR 2015.
- [LF-MMI] D. Povey, et al., "Purely sequence-trained neural networks for ASR based on lattice-free MMI", INTERSPEECH 2016.
- [CTC-CRF] Xiang&Ou. "CRF-based Single-stage Acoustic Modeling with CTC Topology", ICASSP, 2019.

CTC: introducing **blank** symbol

- Motivation: training $p(\mathbf{y} | \mathbf{x})$ without the need for frame-level alignments between the acoustics \mathbf{x} and the transcripts \mathbf{y}
 - Introduce a state sequence $\boldsymbol{\pi} \triangleq \pi_1, \dots, \pi_T$, where $\pi_t \in \text{the-alphabet-of-labels} \cup \langle \mathbf{b} \rangle$



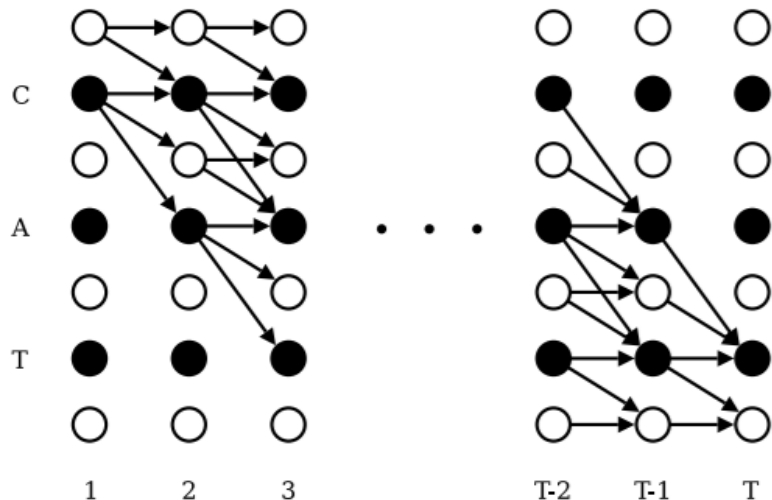
CTC topology

- State topology refers to the state transition structure in π , which basically determines the mapping \mathcal{B}_{CTC} from π to \mathbf{y}

CTC topology : a mapping \mathcal{B}_{CTC} maps π to \mathbf{y} by

- reducing repetitive symbols to a single symbol;
- removing all blank symbols.

$$\mathcal{B}(-CC - -AA - T -) = CAT$$



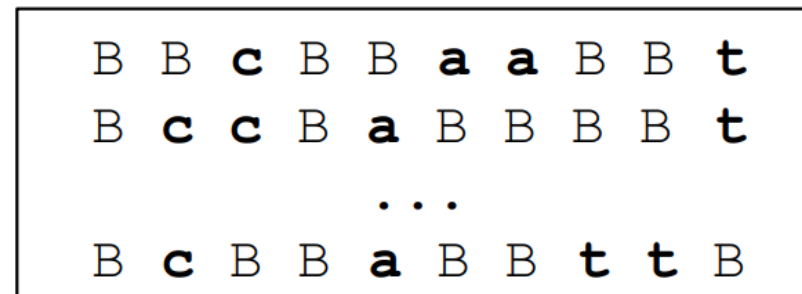
Path posterior

$$p(\boldsymbol{\pi}|\mathbf{x}) = \prod_{t=1}^T p(\pi_t|\mathbf{x})$$

Label-seq posterior

$$p(\mathbf{y}|\mathbf{x}) = \sum_{\boldsymbol{\pi}: \mathcal{B}_{CTC}(\boldsymbol{\pi})=\mathbf{y}} p(\boldsymbol{\pi}|\mathbf{x})$$

Summing over all possible paths, which map to \mathbf{y}



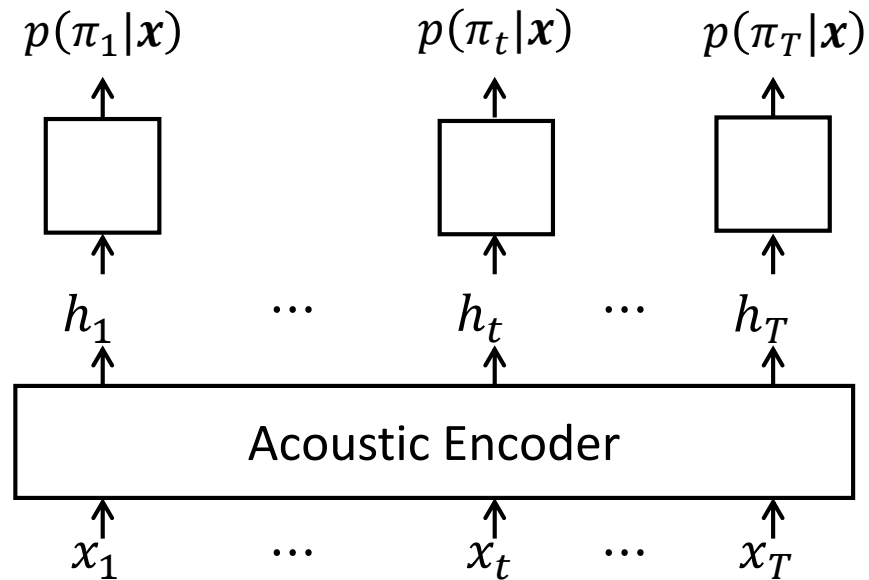
CTC: shortcoming

- Conditional independence assumption

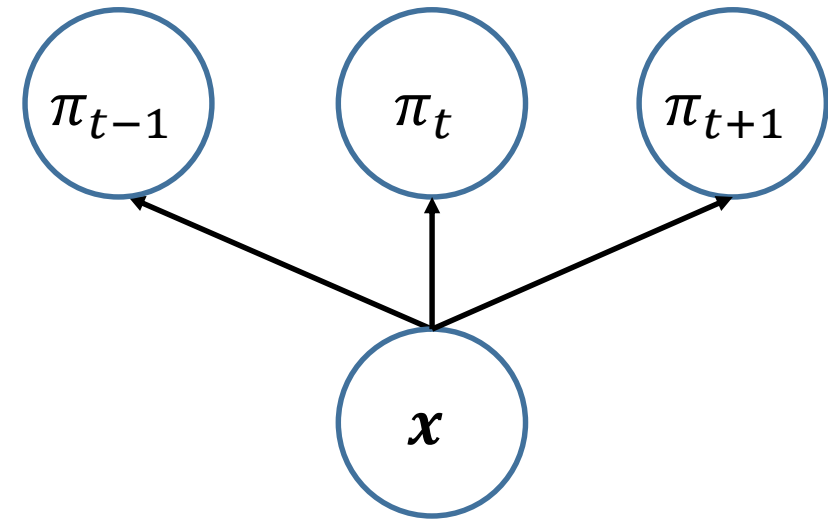
$$p(\boldsymbol{\pi} | \boldsymbol{x}) = \prod_{t=1}^T p(\pi_t | \boldsymbol{x})$$

Overcome

RNN-T
CTC-CRF



Computational flow



Graphical Model Representation

提纲

一、语音识别简史与基础

二、端到端语音识别

三、数据高效

四、多语言与跨语言语音识别

五、总结及展望

Motivation: data-efficient end2end

- End-to-end system:

- Eliminate the construction of GMM-HMMs and phonetic decision-trees, and can be trained from scratch (**flat-start** or **single-stage**)

- In a more strict/ambitious sense:

- Remove the need for a pronunciation lexicon and, even further, train the acoustic and language models jointly rather than separately
- **Data-hungry**

We need data-efficient end2end speech recognition, which uses a separate language model (LM) with or without a pronunciation lexicon.

- Text corpus for language modeling are cheaply available.
- **Data-efficient**

数据高效 Data-efficient

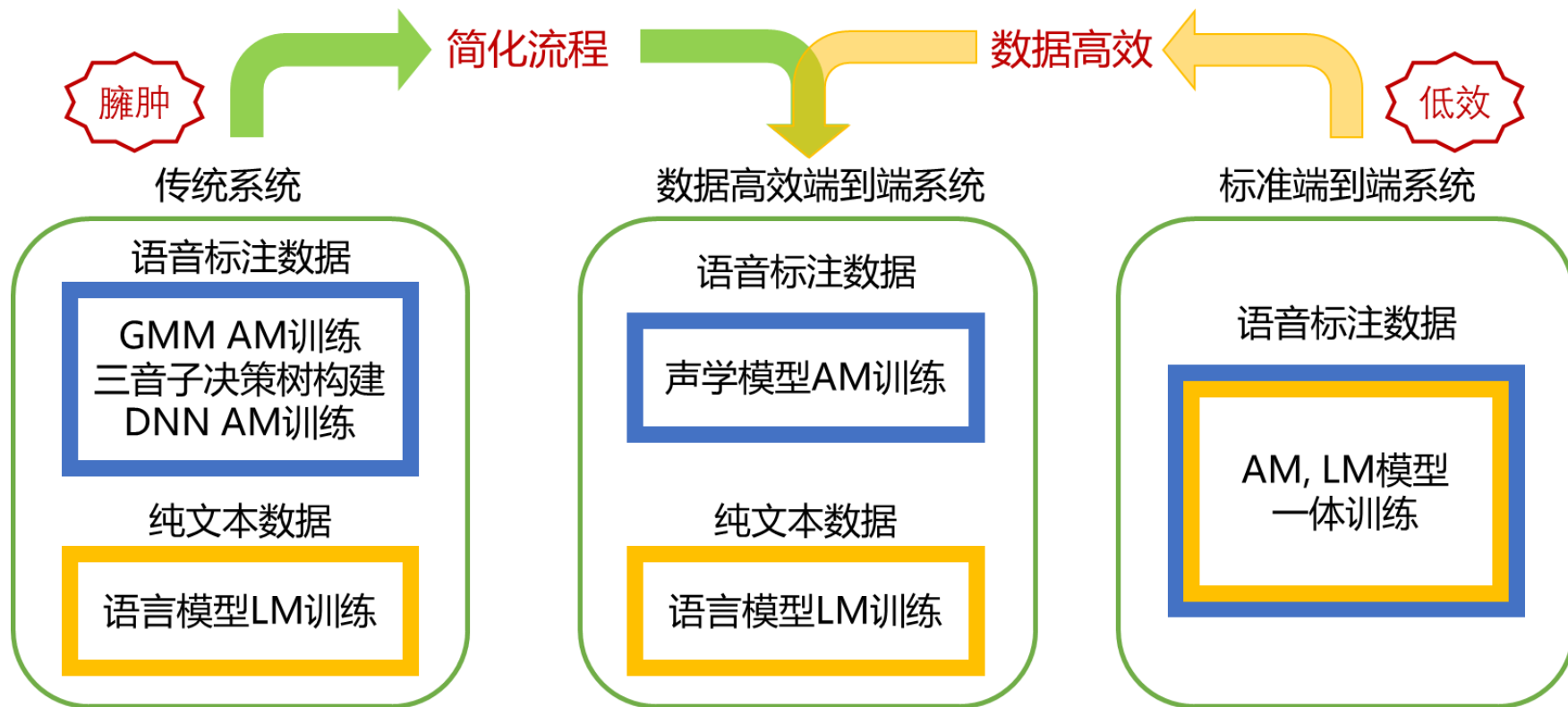
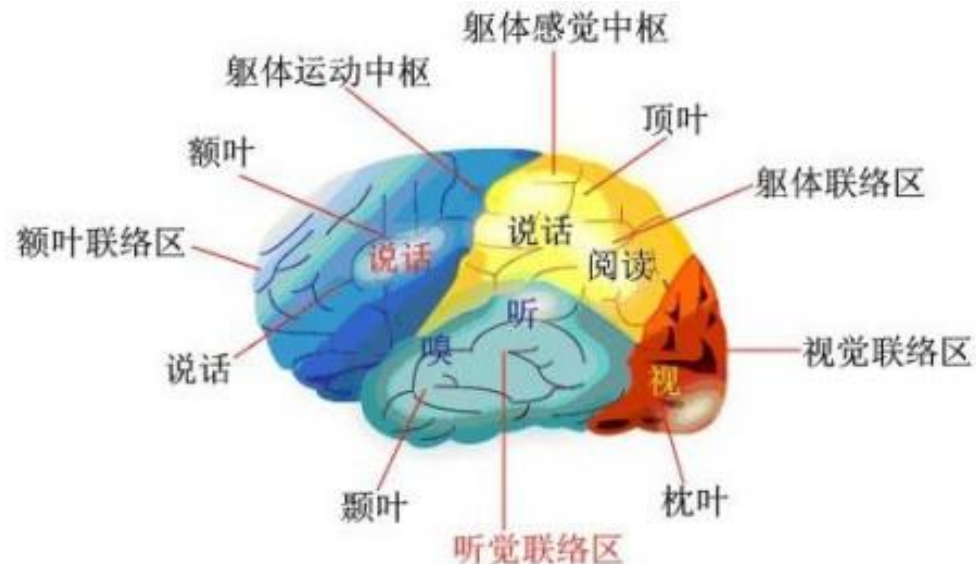
$$\text{效率} = \frac{\text{收益}}{\text{数据人工标注成本}}$$

- 目前语音识别与对话技术，过度依赖有监督学习和大量人工标注数据
- 这里的效率，不是指机器计算的效率（MIPS, million instructions per second），也不是指机器的能耗效率（MIPS/Watt），而是指——**机器学习的效率**
- 谱系化的数据高效的建模与学习方法
 - ✓ 模型架构
 - ✓ 无监督、半监督、自监督学习
 - ✓ 预训练
 - ✓ 迁移学习
 - ✓ 主动学习
 - ✓ 元学习

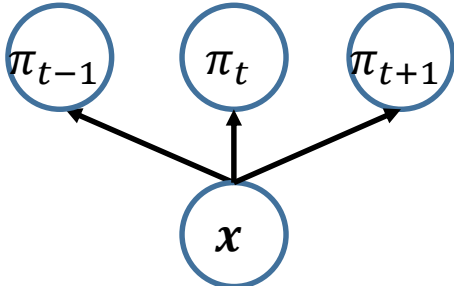
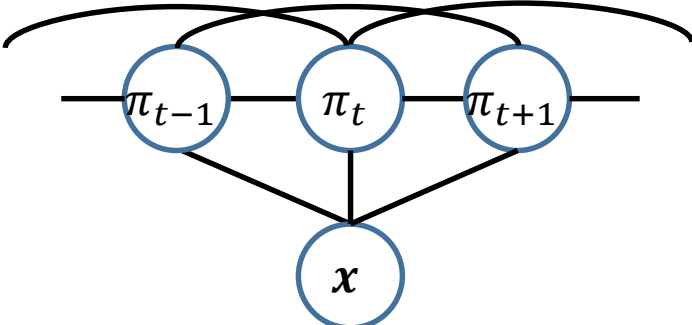
Motivation

适度模块化实现Data-efficiency

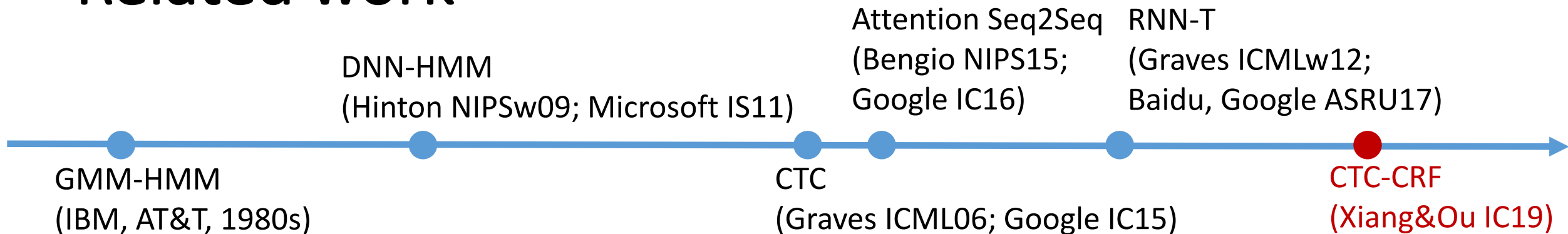
- ✓ 不追求过度的端到端,
- ✓ 保留了声学模型、语言模型的**必要分解**



CTC vs CTC-CRF

CTC	CTC-CRF
$p(\mathbf{y} \mathbf{x}) = \sum_{\boldsymbol{\pi}:\mathcal{B}(\boldsymbol{\pi})=\mathbf{y}} p(\boldsymbol{\pi} \mathbf{x}), \text{ using CTC topology } \mathcal{B}$	
<p>State Independence</p> $p(\boldsymbol{\pi} \mathbf{x}; \boldsymbol{\theta}) = \prod_{t=1}^T p(\pi_t \mathbf{x})$	$p(\boldsymbol{\pi} \mathbf{x}; \boldsymbol{\theta}) = \frac{e^{\phi(\boldsymbol{\pi}, \mathbf{x}; \boldsymbol{\theta})}}{\sum_{\boldsymbol{\pi}'} e^{\phi(\boldsymbol{\pi}', \mathbf{x}; \boldsymbol{\theta})}}$ <p> $\phi(\boldsymbol{\pi}, \mathbf{x}; \boldsymbol{\theta}) = \sum_{t=1}^T \log p(\pi_t \mathbf{x}) + \log p_{LM}(\mathcal{B}(\boldsymbol{\pi}))$ </p> <p> ← Node potential, by NN ← Edge potential, by n-gram denominator LM of labels, like in LF-MMI </p>
$\frac{\partial \log p(\mathbf{y} \mathbf{x}; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \mathbb{E}_{p(\boldsymbol{\pi} \mathbf{y}, \mathbf{x}; \boldsymbol{\theta})} \left[\frac{\partial \log p(\boldsymbol{\pi} \mathbf{x}; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]$	$\frac{\partial \log p(\mathbf{y} \mathbf{x}; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \mathbb{E}_{p(\boldsymbol{\pi} \mathbf{x}, \mathbf{y}; \boldsymbol{\theta})} \left[\frac{\partial \phi(\boldsymbol{\pi}, \mathbf{x}; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right] - \mathbb{E}_{p(\boldsymbol{\pi}' \mathbf{x}; \boldsymbol{\theta})} \left[\frac{\partial \phi(\boldsymbol{\pi}', \mathbf{x}; \boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \right]$
	

Related work



GMM-HMM
(IBM, AT&T, 1980s)

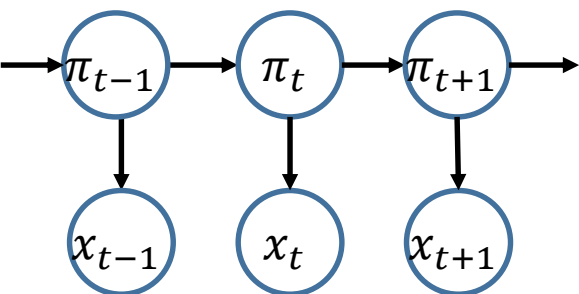
DNN-HMM
(Hinton NIPSw09; Microsoft IS11)

Attention Seq2Seq
(Bengio NIPS15; Google IC16)

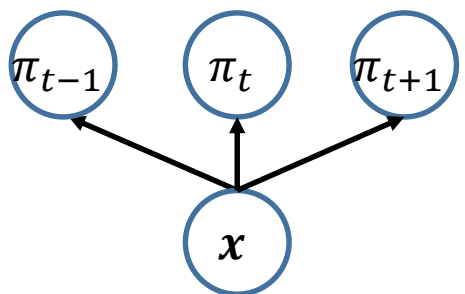
RNN-T
(Graves ICMLw12; Baidu, Google ASRU17)

CTC
(Graves ICML06; Google IC15)

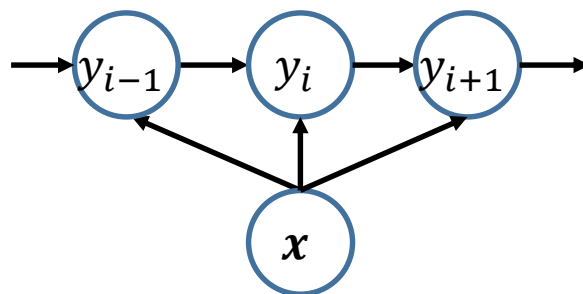
CTC-CRF
(Xiang&Ou IC19)



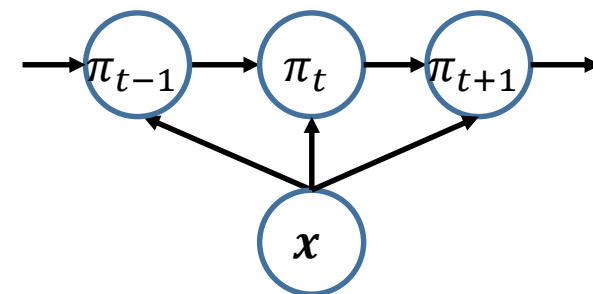
DNN-HMM
缺陷: 多阶段



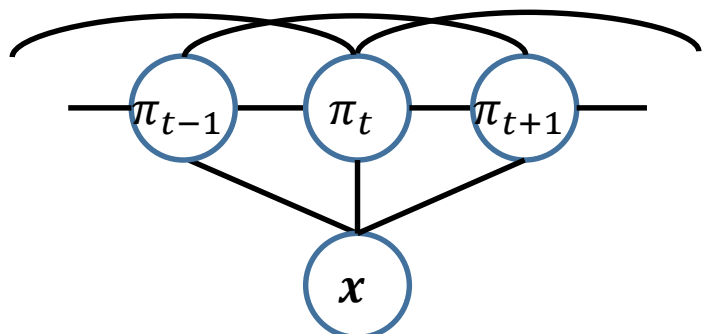
CTC
缺陷: $\{\pi_t\}$ 条件独立性



Attention
缺陷: $\{y_i\}$ 有向图序列模型



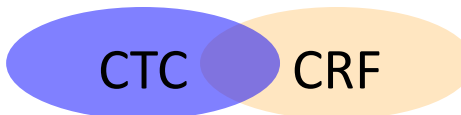
RNN-T
缺陷: $\{\pi_t\}$ 有向图序列模型



提出CTC-CRF

历史上各类模型具有不同的图结构，CTC-CRF占有独特位置！

首次成功地联合神经网络与无向图模型用于语音识别，并在原理上克服了历史上各类模型的不足！

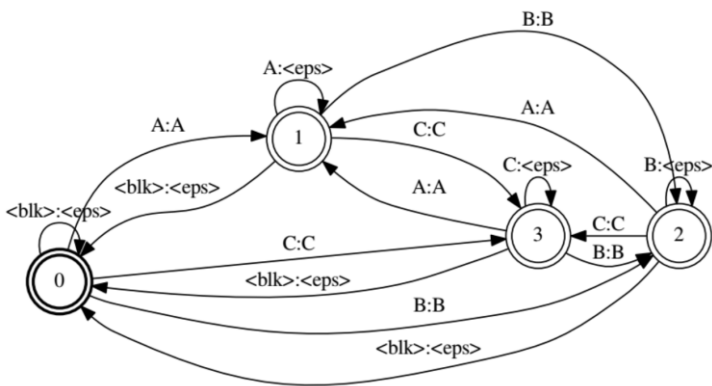


Decoding: LM integration with WFSTs

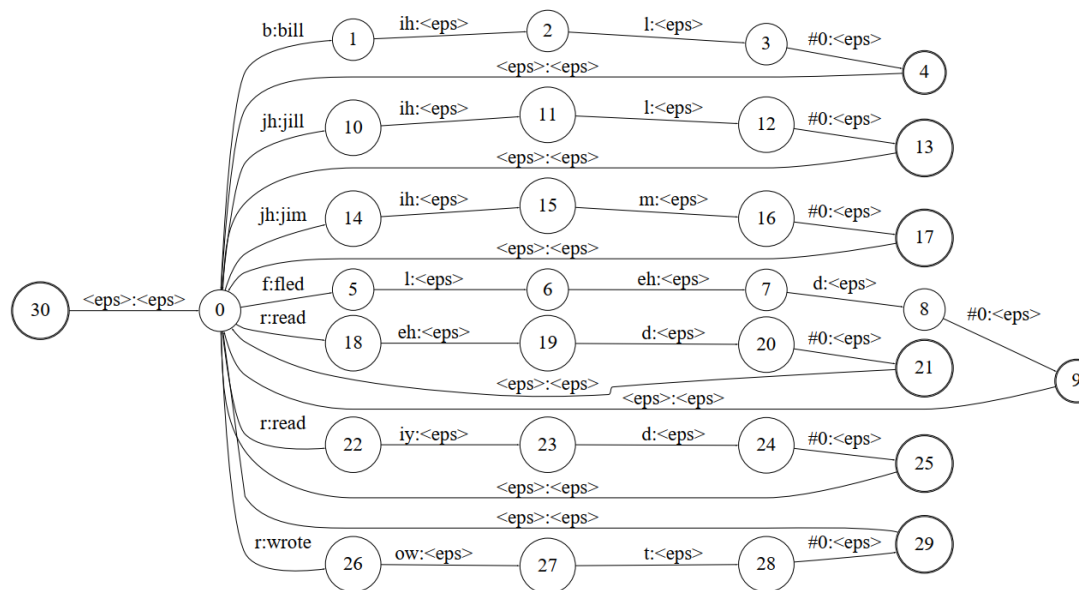
- Best-path-decoding $\max_{\pi} p(\pi|\mathbf{x})$
- Prefix-search-decoding $\max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x})$
- Incorporate lexicon and LM to improve best-path-decoding

$$\max_{\pi} p(\pi|\mathbf{x}) LM_{External}(\mathcal{B}_{CTC}(\pi))$$

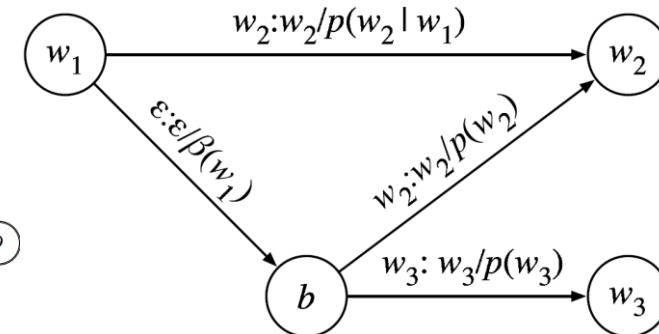
WFST representing CTC topology: T



Lexicon: L



Language model: G



Composed and optimized into a single WFST

Experiments

- We conduct our experiments on three benchmark datasets:
 - WSJ 80 hours
 - Switchboard 300 hours
 - Librispeech 1000 hours
- **Acoustic model:** 6 layer BLSTM with **320** hidden dim, 13M parameters
- **Adam optimizer** with an initial learning rate of 0.001, decreased to 0.0001 when cv loss does not decrease
- **Implemented with Pytorch.**
- **Objective function** (use the CTC objective function to help convergences):

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

- **Decoding score function** (use word-based language models, WFST based decoding):

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

Experiments (Comparison with CTC, phone based)

WSJ 80h

Model	Unit	LM	SP	dev93	eval92
CTC	Mono-phone	4-gram	N	10.81%	7.02%
CTC-CRF	Mono-phone	4-gram	N	6.24%	3.90%

44.4% reduction in eval92 error rate for CTC-CRF compared to CTC.

Switchboard 300h

Model	Unit	LM	SP	SW	CH
CTC	Mono-phone	4-gram	N	12.9%	23.6%
CTC-CRF	Mono-phone	4-gram	N	11.0%	21.0%

14.7% reduction in SW error rate and 11% reduction in CH error rate for CTC-CRF compared to CTC.

Librispeech 1000h

Model	Unit	LM	SP	Dev Clean	Dev Other	Test Clean	Test Other
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%

19.1% reduction in Test Clean error rate and 22.1% reduction in Test Other error rate for CTC-CRF compared to CTC.

SP: speed perturbation for 3-fold data augmentation.

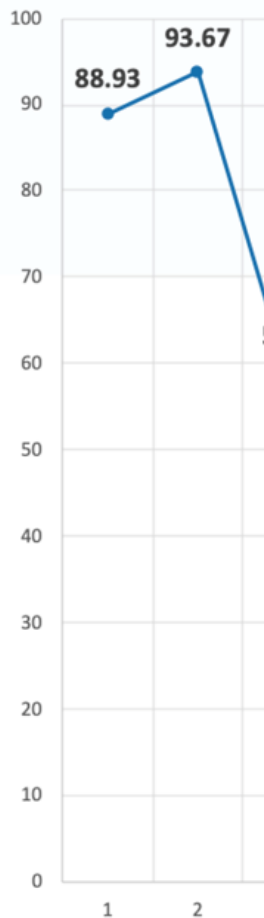
2021 SLT CHILDREN SPEECH RECOGNITION CHALLENGE (CSRC)

ORGANIZER :  西北工业大学  清华大学  厦門大學  标贝科技 

- 400 hours of data, targeting to boost children speech recognition research.
- Evaluated on 10 hours of children's reading and conversational speech.
- 3 baselines (Chain model, Transformer and CTC-CRF) are provided.

model	Chain model	Transformer	CTC-CRF
CER%	28.75	27.28	25.34

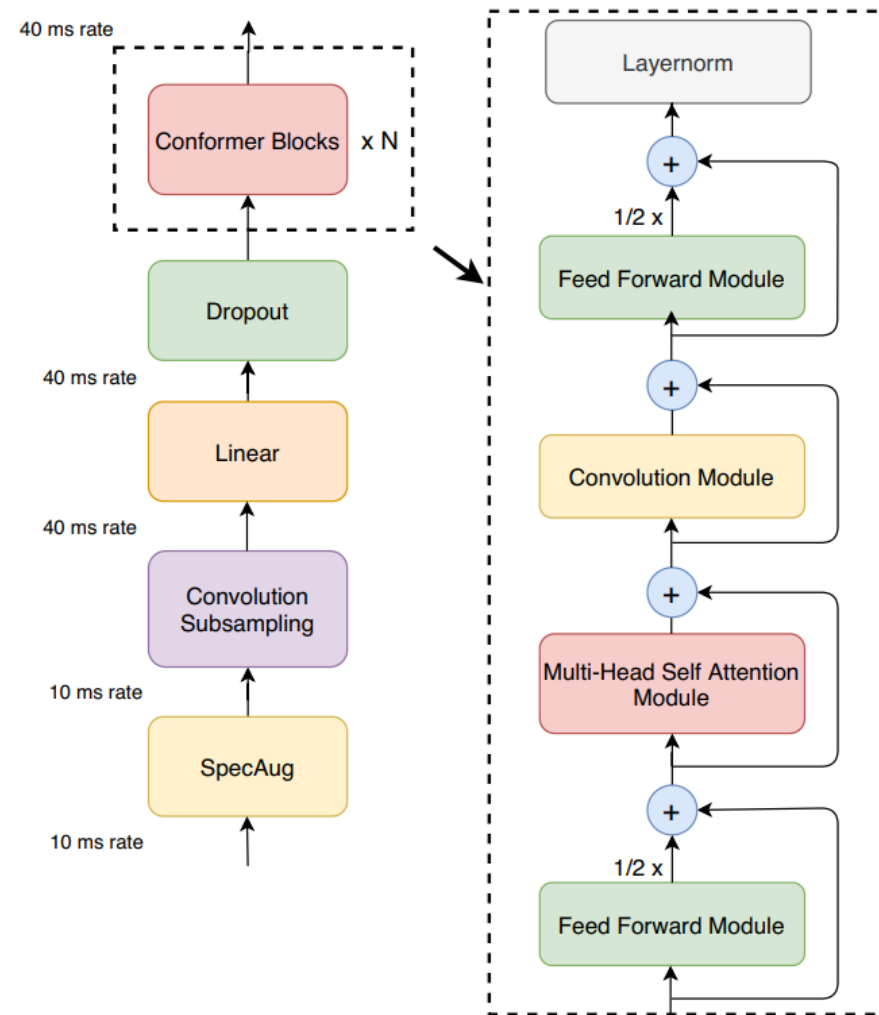
Fan Yu, Zhuoyuan Yao, Xiong Wang, Keyu An, Lei Xie, **Zhijian Ou**, Bo Liu, Xiulin Li, Guanqiong Miao. The SLT 2021 children speech recognition challenge: Open datasets, rules and baselines. SLT 2021.



Advancing CTC-CRF Based End-to-End Speech Recognition with Wordpieces and Conformers

Huahuan Zheng, Wenjie Peng, **Zhijian Ou** and Jinsong Zhang, arXiv:2107.03007

Basic Units of Labels	Label Sequence
phoneme	DH AE1 T N IY1 DH ER0 AH1 V DH EH1 M HH AE1 D K R AO1 S T DH AH0 TH R EH1 SH OW2 L D S IH1 N S DH AH0 D AA1 R K D EY1
character /grapheme	that_neither_of_them_had_crossed_the_threshold_since_the_dark_day_
subword /wordpiece	that_neither_of_them_had_crossed_the_ threshold_since_the_dark_day_
word	that neither of them had crossed the threshold since the dark day



Experiments (Comparison with SOTA)

Mandarin Aishell 170h %CER

Model	%CER
LF-MMI with i-vector [1]	7.43
Transformer [2]	6.7
CTC-CRF [3]	6.34
CTC-CRF (3-gram LM)	4.90
RNN-T by our implementation	4.82
U2 ++ [4]	5.19
K2, conformer MMI [5]	4.94

[1] D. Povey, A. Ghoshal, and et al, “The Kaldi speech recognition toolkit,” ASRU 2011.

[2] S. Karita, N. Chen, and et al, “A comparative study on transformer vs RNN in speech applications,” ASRU 2019.

[3] Keyu An, Hongyu Xiang, and Zhijian Ou, “CAT: A CTC-CRF based ASR toolkit bridging the hybrid and the end-to-end approaches towards data efficiency and low latency,” INTERSPEECH 2020.

[4] U2++: Unified Two-pass Bidirectional End-to-end Model for Speech Recognition, arXiv 2106.05642

[5] <https://github.com/k2-fsa/icefall/blob/master/egs/aishell/ASR/RESULTS.md>

Experiments (Comparison with SOTA)

English Switchboard 300h %WER

Model	#params	LM	unit	SW	CH	Eval2000
RNN-T, 2021 [10]	57	RNN LM	char	6.4	13.4	9.9
Att. Conformer [9]	44.6	Trans.	bpe	6.8	14.0	10.4
TDNN-F [11]	-	Trans.*	triphone	7.2	14.4	10.8
TDNN-F [11]	-	Trans.**	triphone	6.5	13.9	10.2
VGGBLSTM [2]	39.15	RNN LM	monophone	8.8	17.4	[13.0]
Conformer	51.82	Trans.	monophone	6.9	14.5	10.7
(This work)	51.85	Trans.	wordpiece	7.2	14.8	11.1

* N-best rescoring, ** Iterative lattice rescoring

[2] “CAT: A CTC-CRF based ASR toolkit bridging the hybrid and the end-to-end approaches towards data efficiency and low latency,” INTERSPEECH 2020.

[9] “Conformer: Convolution-augmented Transformer for Speech Recognition”, Interspeech 2020.

[10] “Advancing RNN transducer technology for speech recognition,” ICASSP 2021.

[11] “A parallelizable lattice rescoring strategy with neural language models,” ICASSP, 2021

Experiment results

- The CTC-CRF framework inherits the **data-efficiency** of the hybrid approach and the **simplicity** of the end-to-end approach.
- CTC-CRF significantly **outperforms** regular CTC on a wide range of benchmarks, and is **on par with** other state-of-the-art end-to-end models.
 - English WSJ-80h, Switchboard-300h, Librispeech-1000h; Mandarin Aishell-170h; Hokkien 100h; ...
- **Flexibility**
 - Streaming ASR <- INTRESPEECH 2020
 - Neural Architecture Search <- SLT 2021
 - Children Speech Recognition <- SLT 2021
 - Wordpieces, Conformer architectures
 - Multilingual and Crosslingual <- ASRU2021
 - ...



<https://github.com/thu-spmi/cat>

提纲

一、语音识别简史与基础

二、端到端语音识别

三、数据高效

四、多语言与跨语言语音识别

五、总结及展望

Section Content

1. Motivation

2. Related work

3. Method: **JoinAP**

4. Experiments

5. Conclusion

- Chengrui Zhu, Keyu An, Huahuan Zheng, Zhijian Ou. “Multilingual and Crosslingual Speech Recognition using Phonological-Vector based Phone Embeddings”, ASRU 2021.

Motivation

- There are more than 7100 languages in the world, and most of them are low-resourced languages.
- Multilingual speech recognition
 - Training data from a number of languages (seen languages) are merged to train a multilingual AM.
- Crosslingual speech recognition
 - The target language is unseen in training the multilingual AM.
 - In **few-shot** setting , the AM can be finetuned on limited target language data.
 - In **zero-shot** setting , the AM is directly used without finetuning*.

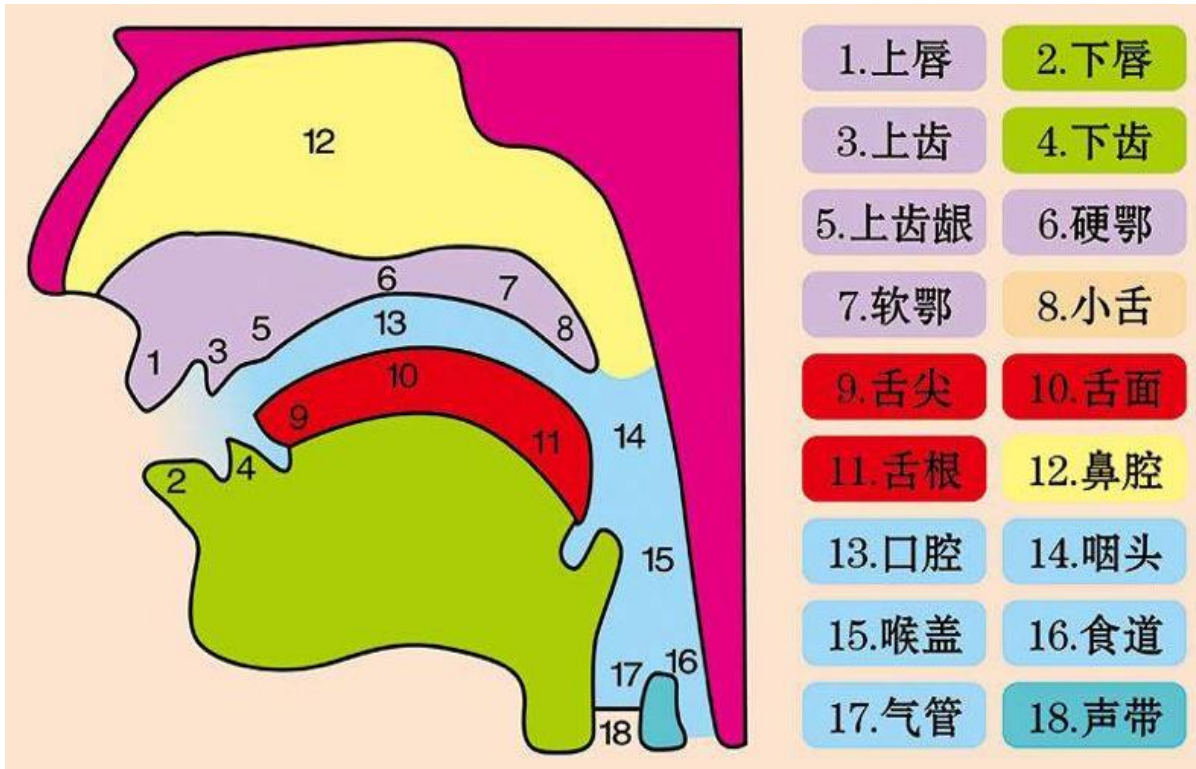
* Suppose that text corpus from the target language are available.

Intuitively, the key to successful multilingual and crosslingual recognition is to promote the information sharing in multilingual training and maximize the knowledge transferring from the well trained multilingual model to the model for recognizing the utterances in the new language.

Universal Phone Set

- International Phonetic Alphabet (IPA), 1888

无论哪种人类语言，都是人类的一套发音器官发出来的音



THE INTERNATIONAL PHONETIC ALPHABET (revised to 2020)

CONSONANTS (PULMONIC)

© 2020 IPA

	Bilabial	Labiodental	Dental	Alveolar	Postalveolar	Retroflex	Palatal	Velar	Uvular	Pharyngeal	Glottal
Plosive	p b			t d		ʈ ɖ	c ɟ	k ɡ	q ɢ		ʔ
Nasal	m	ɱ		n		ɳ	ɲ	ŋ	ɴ		
Trill				r					ʀ		
Tap or Flap		ɸ		ɾ		ɽ					
Fricative	ɸ β	f v	θ ð	s z	ʃ ʒ	ʂ ʐ	ç ʝ	x ɣ	χ ʁ	ħ ʕ	h ɦ
Lateral fricative				ɬ ɮ							
Approximant		ʋ		ɹ		ɻ	j	ɰ			
Lateral approximant				l		ɭ	ʎ	ʟ			

Symbols to the right in a cell are voiced, to the left are voiceless. Shaded areas denote articulations judged impossible.

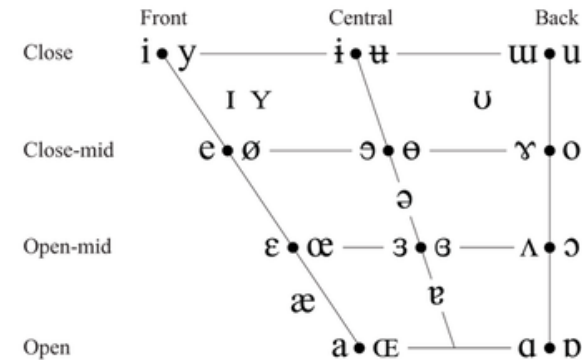
CONSONANTS (NON-PULMONIC)

Clicks	Voiced implosives	Ejectives
◌ Bilabial	ɓ Bilabial	ʼ Examples:
Dental	ɗ Dental/alveolar	pʼ Bilabial
! (Post)alveolar	ɟ Palatal	tʼ Dental/alveolar
‡ Palatoalveolar	ɠ Velar	kʼ Velar
Alveolar lateral	ɣ Uvular	sʼ Alveolar fricative

OTHER SYMBOLS

- ʍ Voiceless labial-velar fricative
- ʋ Voiced labial-velar approximant
- ɥ Voiced labial-palatal approximant
- ħ Voiceless epiglottal fricative
- ʕ Voiced epiglottal fricative
- ʡ Epiglottal plosive
- ɕ ʑ Alveolo-palatal fricatives
- ɺ Voiced alveolar lateral flap
- ɥ̟ Simultaneous ʃ and x
- Affricates and double articulations can be represented by two symbols joined by a tie bar if necessary.

VOWELS



Where symbols appear in pairs, the one to the right represents a rounded vowel.

SUPRASEGMENTALS

- ˈ Primary stress
- ˌ Secondary stress
- ː Long
- ˑ Half-long
- ◌̥ Extra-short

DIACRITICS

Symbol	Description
◌̥	Extra-short
◌̇	Shortening diacritic
◌̈	Shortening diacritic
◌̉	Shortening diacritic
◌̊	Shortening diacritic
◌̋	Shortening diacritic
◌̌	Shortening diacritic
◌̍	Shortening diacritic
◌̎	Shortening diacritic
◌̏	Shortening diacritic
◌̐	Shortening diacritic
◌̑	Shortening diacritic
◌̒	Shortening diacritic
◌̓	Shortening diacritic
◌̔	Shortening diacritic
◌̕	Shortening diacritic
◌̖	Shortening diacritic
◌̗	Shortening diacritic
◌̘	Shortening diacritic
◌̙	Shortening diacritic
◌̚	Shortening diacritic
◌̜	Shortening diacritic
◌̝	Shortening diacritic
◌̞	Shortening diacritic
◌̟	Shortening diacritic
◌̠	Shortening diacritic
◌̡	Shortening diacritic
◌̢	Shortening diacritic
◌̣	Shortening diacritic
◌̤	Shortening diacritic
◌̥	Shortening diacritic
◌̦	Shortening diacritic
◌̧	Shortening diacritic
◌̨	Shortening diacritic
◌̩	Shortening diacritic
◌̪	Shortening diacritic
◌̫	Shortening diacritic
◌̬	Shortening diacritic
◌̭	Shortening diacritic
◌̮	Shortening diacritic
◌̯	Shortening diacritic
◌̰	Shortening diacritic
◌̱	Shortening diacritic
◌̲	Shortening diacritic
◌̳	Shortening diacritic
◌̴	Shortening diacritic
◌̵	Shortening diacritic
◌̶	Shortening diacritic
◌̷	Shortening diacritic
◌̸	Shortening diacritic
◌̹	Shortening diacritic
◌̺	Shortening diacritic
◌̻	Shortening diacritic
◌̼	Shortening diacritic
◌̽	Shortening diacritic
◌̾	Shortening diacritic
◌̿	Shortening diacritic
◌̀	Shortening diacritic
◌́	Shortening diacritic
◌̂	Shortening diacritic
◌̃	Shortening diacritic
◌̄	Shortening diacritic
◌̅	Shortening diacritic
◌̆	Shortening diacritic
◌̇	Shortening diacritic
◌̈	Shortening diacritic
◌̉	Shortening diacritic
◌̊	Shortening diacritic
◌̋	Shortening diacritic
◌̌	Shortening diacritic
◌̍	Shortening diacritic
◌̎	Shortening diacritic
◌̏	Shortening diacritic
◌̐	Shortening diacritic
◌̑	Shortening diacritic
◌̒	Shortening diacritic
◌̓	Shortening diacritic
◌̔	Shortening diacritic
◌̕	Shortening diacritic
◌̖	Shortening diacritic
◌̗	Shortening diacritic
◌̘	Shortening diacritic
◌̙	Shortening diacritic
◌̚	Shortening diacritic
◌̜	Shortening diacritic
◌̝	Shortening diacritic
◌̞	Shortening diacritic
◌̟	Shortening diacritic
◌̠	Shortening diacritic
◌̡	Shortening diacritic
◌̢	Shortening diacritic
◌̣	Shortening diacritic
◌̤	Shortening diacritic
◌̥	Shortening diacritic
◌̦	Shortening diacritic
◌̧	Shortening diacritic
◌̨	Shortening diacritic
◌̩	Shortening diacritic
◌̪	Shortening diacritic
◌̫	Shortening diacritic
◌̬	Shortening diacritic
◌̭	Shortening diacritic
◌̮	Shortening diacritic
◌̯	Shortening diacritic
◌̰	Shortening diacritic
◌̱	Shortening diacritic
◌̲	Shortening diacritic
◌̳	Shortening diacritic
◌̴	Shortening diacritic
◌̵	Shortening diacritic
◌̶	Shortening diacritic
◌̷	Shortening diacritic
◌̸	Shortening diacritic
◌̹	Shortening diacritic
◌̺	Shortening diacritic
◌̻	Shortening diacritic
◌̼	Shortening diacritic
◌̽	Shortening diacritic
◌̾	Shortening diacritic
◌̿	Shortening diacritic
◌̀	Shortening diacritic
◌́	Shortening diacritic
◌̂	Shortening diacritic
◌̃	Shortening diacritic
◌̄	Shortening diacritic
◌̅	Shortening diacritic
◌̆	Shortening diacritic
◌̇	Shortening diacritic
◌̈	Shortening diacritic
◌̉	Shortening diacritic
◌̊	Shortening diacritic
◌̋	Shortening diacritic
◌̌	Shortening diacritic
◌̍	Shortening diacritic
◌̎	Shortening diacritic
◌̏	Shortening diacritic
◌̐	Shortening diacritic
◌̑	Shortening diacritic
◌̒	Shortening diacritic
◌̓	Shortening diacritic
◌̔	Shortening diacritic
◌̕	Shortening diacritic
◌̖	Shortening diacritic
◌̗	Shortening diacritic
◌̘	Shortening diacritic
◌̙	Shortening diacritic
◌̚	Shortening diacritic
◌̜	Shortening diacritic
◌̝	Shortening diacritic
◌̞	Shortening diacritic
◌̟	Shortening diacritic
◌̠	Shortening diacritic
◌̡	Shortening diacritic
◌̢	Shortening diacritic
◌̣	Shortening diacritic
◌̤	Shortening diacritic
◌̥	Shortening diacritic
◌̦	Shortening diacritic
◌̧	Shortening diacritic
◌̨	Shortening diacritic
◌̩	Shortening diacritic
◌̪	Shortening diacritic
◌̫	Shortening diacritic
◌̬	Shortening diacritic
◌̭	Shortening diacritic
◌̮	Shortening diacritic
◌̯	Shortening diacritic
◌̰	Shortening diacritic
◌̱	Shortening diacritic
◌̲	Shortening diacritic
◌̳	Shortening diacritic
◌̴	Shortening diacritic
◌̵	Shortening diacritic
◌̶	Shortening diacritic
◌̷	Shortening diacritic
◌̸	Shortening diacritic
◌̹	Shortening diacritic
◌̺	Shortening diacritic
◌̻	Shortening diacritic
◌̼	Shortening diacritic
◌̽	Shortening diacritic
◌̾	Shortening diacritic
◌̿	Shortening diacritic
◌̀	Shortening diacritic
◌́	Shortening diacritic
◌̂	Shortening diacritic
◌̃	Shortening diacritic
◌̄	Shortening diacritic
◌̅	Shortening diacritic
◌̆	Shortening diacritic
◌̇	Shortening diacritic
◌̈	Shortening diacritic
◌̉	Shortening diacritic
◌̊	Shortening diacritic
◌̋	Shortening diacritic
◌̌	Shortening diacritic
◌̍	Shortening diacritic
◌̎	Shortening diacritic
◌̏	Shortening diacritic
◌̐	Shortening diacritic
◌̑	Shortening diacritic
◌̒	Shortening diacritic
◌̓	Shortening diacritic
◌̔	Shortening diacritic
◌̕	Shortening diacritic
◌̖	Shortening diacritic
◌̗	Shortening diacritic
◌̘	Shortening diacritic
◌̙	Shortening diacritic
◌̚	Shortening diacritic
◌̜	Shortening diacritic
◌̝	Shortening diacritic
◌̞	Shortening diacritic
◌̟	Shortening diacritic
◌̠	Shortening diacritic
◌̡	Shortening diacritic
◌̢	Shortening diacritic
◌̣	Shortening diacritic
◌̤	Shortening diacritic
◌̥	Shortening diacritic
◌̦	Shortening diacritic
◌̧	Shortening diacritic
◌̨	Shortening diacritic
◌̩	Shortening diacritic
◌̪	Shortening diacritic
◌̫	Shortening diacritic
◌̬	Shortening diacritic
◌̭	Shortening diacritic
◌̮	Shortening diacritic
◌̯	Shortening diacritic
◌̰	Shortening diacritic
◌̱	Shortening diacritic
◌̲	Shortening diacritic
◌̳	Shortening diacritic
◌̴	Shortening diacritic
◌̵	Shortening diacritic
◌̶	Shortening diacritic
◌̷	Shortening diacritic
◌̸	Shortening diacritic
◌̹	Shortening diacritic
◌̺	Shortening diacritic
◌̻	Shortening diacritic
◌̼	Shortening diacritic
◌̽	Shortening diacritic
◌̾	Shortening diacritic
◌̿	Shortening diacritic
◌̀	Shortening diacritic
◌́	Shortening diacritic
◌̂	Shortening diacritic
◌̃	Shortening diacritic
◌̄	Shortening diacritic
◌̅	Shortening diacritic
◌̆	Shortening diacritic
◌̇	Shortening diacritic
◌̈	Shortening diacritic
◌̉	Shortening diacritic
◌̊	Shortening diacritic
◌̋	Shortening diacritic
◌̌	Shortening diacritic
◌̍	Shortening diacritic
◌̎	Shortening diacritic
◌̏	Shortening diacritic
◌̐	Shortening diacritic
◌̑	Shortening diacritic
◌̒	Shortening diacritic
◌̓	Shortening diacritic
◌̔	Shortening diacritic
◌̕	Shortening diacritic
◌̖	Shortening diacritic
◌̗	Shortening diacritic
◌̘	Shortening diacritic
◌̙	Shortening diacritic
◌̚	Shortening diacritic
◌̜	Shortening diacritic
◌̝	Shortening diacritic
◌̞	Shortening diacritic
◌̟	Shortening diacritic
◌̠	Shortening diacritic
◌̡	Shortening diacritic
◌̢	Shortening diacritic
◌̣	Shortening diacritic
◌̤	Shortening diacritic
◌̥	Shortening diacritic
◌̦	Shortening diacritic
◌̧	Shortening diacritic
◌̨	Shortening diacritic
◌̩	Shortening diacritic
◌̪	Shortening diacritic
◌̫	Shortening diacritic
◌̬	Shortening diacritic
◌̭	Shortening diacritic
◌̮	Shortening diacritic
◌̯	Shortening diacritic
◌̰	Shortening diacritic
◌̱	Shortening diacritic
◌̲	Shortening diacritic
◌̳	Shortening diacritic
◌̴	Shortening diacritic
◌̵	Shortening diacritic
◌̶	Shortening diacritic
◌̷	Shortening diacritic
◌̸	Shortening diacritic
◌̹	Shortening diacritic
◌̺	Shortening diacritic
◌̻	Shortening diacritic
◌̼	Shortening diacritic
◌̽	Shortening diacritic
◌̾	Shortening diacritic
◌̿	Shortening diacritic
◌̀	Shortening diacritic
◌́	Shortening diacritic
◌̂	Shortening diacritic
◌̃	Shortening diacritic
◌̄	Shortening diacritic
◌̅	Shortening diacritic
◌̆	Shortening diacritic
◌̇	Shortening diacritic
◌̈	Shortening diacritic
◌̉	Shortening diacritic
◌̊	Shortening diacritic
◌̋	Shortening diacritic
◌̌	Shortening diacritic
◌̍	Shortening diacritic
◌̎	Shortening diacritic
◌̏	Shortening diacritic
◌̐	Shortening diacritic
◌̑	Shortening diacritic
◌̒	Shortening diacritic
◌̓	Shortening diacritic
◌̔	Shortening diacritic
◌̕	Shortening diacritic
◌̖	Shortening diacritic
◌̗	Shortening diacritic
◌̘	Shortening diacritic
◌̙	Shortening diacritic
◌̚	Shortening diacritic
◌̜	Shortening diacritic
◌̝	Shortening diacritic
◌̞	Shortening diacritic
◌̟	Shortening diacritic
◌̠	Shortening diacritic
◌̡	Shortening diacritic
◌̢	Shortening diacritic
◌̣	Shortening diacritic
◌̤	Shortening diacritic
◌̥	Shortening diacritic
◌̦	Shortening diacritic
◌̧	Shortening diacritic
◌̨	Shortening diacritic
◌̩	Shortening diacritic
◌̪	Shortening diacritic
◌̫	Shortening diacritic
◌̬	Shortening diacritic

Phonological features

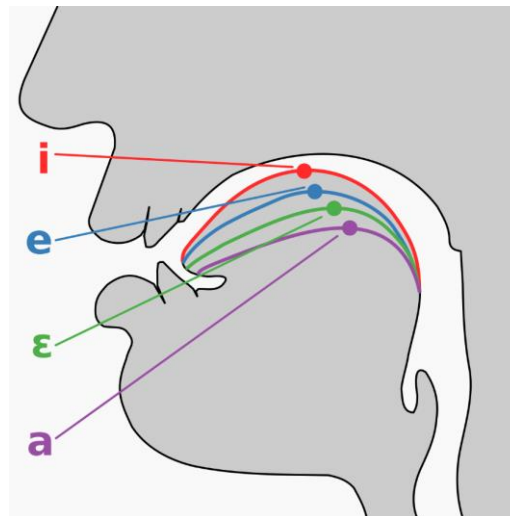
- Often **phones** are seen as being the “atoms” of speech.
- But it is now widely accepted in phonology that phones are decomposable into smaller, more fundamental units, sharable across all languages, called **phonological (distinctive) features**.
- Describe phones by phonological features

- Vowels

- vowel height
 - vowel backness

- Consonants

- Place of articulation
 - Manner of articulation



Phonological feature	d	ε	ð	ə	i	ɖ	kʲ
syllabic	-	+	-	+	+	-	-
sonorant	-	+	-	+	+	-	-
consonantal	+	-	+	-	-	+	+
continuant	-	+	+	+	+	-	-
delayed release	-	-	-	-	-	+	-
lateral	-	-	-	-	-	-	-
nasal	-	-	-	-	-	-	-
strident	0	0	0	0	0	0	0
voice	+	+	+	+	+	+	-
spread glottis	-	-	-	-	-	-	-
constricted glottis	-	-	-	-	-	-	-
anterior	+	0	+	0	0	-	-
coronal	+	-	+	-	-	+	-
distributed labial	-	0	+	0	0	+	0
labial	-	-	-	-	-	-	-
high	-	-	-	-	+	+	+
low	-	-	-	-	-	-	-
back	-	-	-	+	-	-	-
round	-	-	-	-	-	-	-
velaric	-	-	-	-	-	-	-
tense	0	-	0	-	+	0	0
long	-	-	-	-	-	-	-
hitone	0	0	0	0	0	0	0
hireg	0	0	0	0	0	0	0

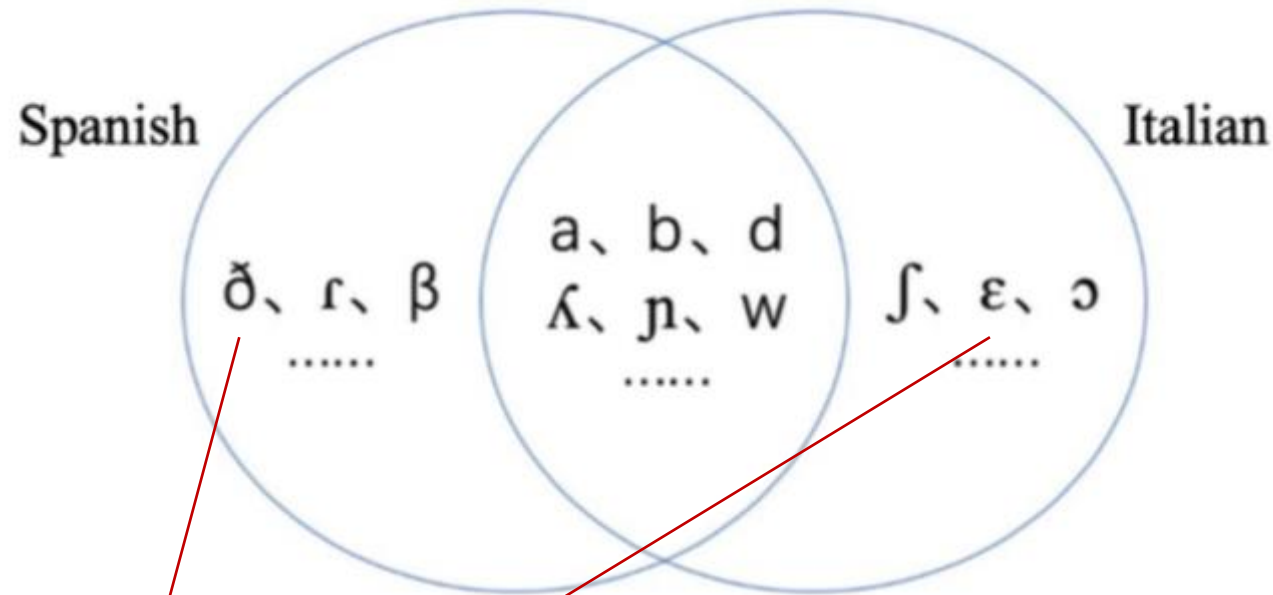
Phonological features: micro-decomposition of phones

- Like atoms could be split into nucleus and electrons, phones can be expressed by phonological features.

Matter	Speech
Atoms	Phones
Periodic table of elements	IPA table
Nucleus, electrons	Phonological features

Phonological features: promote information sharing

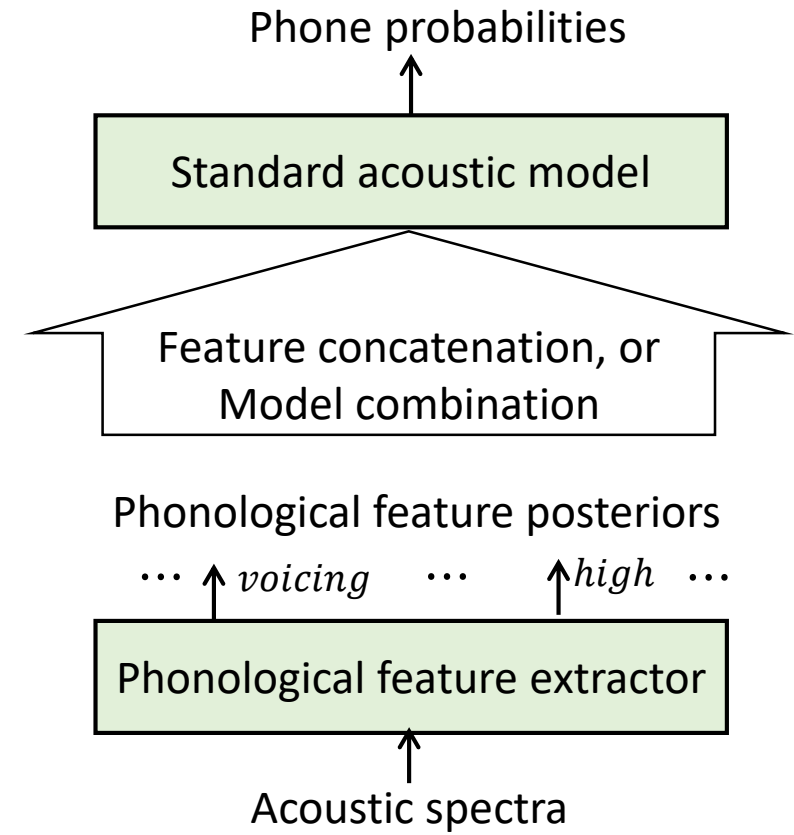
- Even language-specific phones are connected by using phonological features.



ð : -, +, +, -, -, -, 0, +, -, -, +, +, +, -, -, -, -, -, -, 0, -, 0, 0
ε : +, +, -, +, -, -, -, 0, +, -, -, 0, -, 0, -, -, -, +, -, -, +, -, 0, 0

Related work

- Phonological features (PFs) have been applied in multilingual and crosslingual ASR
- Previous studies generally take a bottom-up approach, and suffer from:
 - The acoustic-to-PF extraction in a bottom-up way is itself **difficult**.
 - Do not provide a principled model to calculate the phone probabilities **for unseen phones** from the new language towards zero-shot crosslingual recognition.



From phonological features to phonological-vector

- Phonological-vector

- Encode each phonological feature by a 2-bit binary vector. (24PFs -> 48bits)

+	-	0
10	01	00

- Plus 3 bits to indicate <blk>, <spn>, <nsm>
- Phonological-vector: Total 51 bits

Joining of Acoustics and Phonology (JoinAP)

- The JoinAP method

- DNN based acoustic feature extraction (bottom-up) and phonology driven phone embedding (top-down) are joined to calculate the **logits**.

- JoinAP-Linear

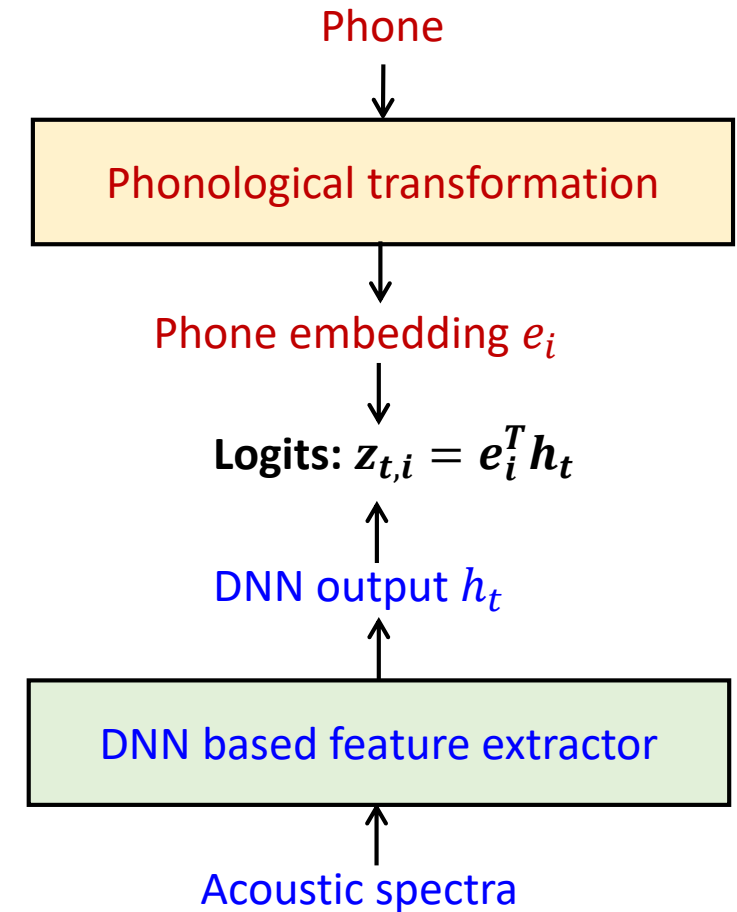
- Linear transformation of phonological-vector p_i to define the embedding vector for phone i :

$$e_i = Ap_i \in \mathbb{R}^H$$

- JoinAP-Nonlinear

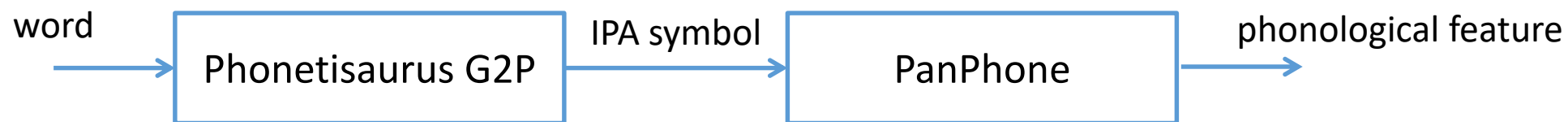
- Apply nonlinear transformation, multilayered neural networks:

$$e_i = A_2 \sigma(A_1 p_i) \in \mathbb{R}^H$$



Experiments

- Train multilingual AM on German, French, Spanish and Polish.
- Zero-shot and few-shot crosslingual ASR on Polish and Mandarin.



- Use CTC-CRF based ASR toolkit, CAT
 - **Acoustic model:** 3 layer VGGBLSTM with **1024** hidden dim
 - **Adam optimizer:** with an initial learning rate of 0.001, decreased to 1/10 until less than 0.00001
 - **Dropout** 0.5

Language	Corpora	#Phones	Train	Dev	Test
German	CommonVoice	40	639.4	24.7	25.1
French	CommonVoice	57	465.2	21.9	23.0
Spanish	CommonVoice	30	246.4	24.9	25.6
Italian	CommonVoice	33	89.3	19.7	20.8
Polish	CommonVoice	46	93.2	5.2	6.1
Mandarin	AISHELL-1	96	150.9	18.1	10.0

Experiments

- Multilingual experiments

Language	Flat-Phone monolingual	Flat-Phone w/o finetuning	Flat-Phone finetuning	JoinAP-Linear w/o finetuning	JoinAP-Linear finetuning	JoinAP-Nonlinear w/o finetuning	JoinAP-Nonlinear finetuning
German	13.09	14.36	12.42	13.72	12.45	13.97	12.64
French	18.96	22.73	18.91	22.73	19.54	22.88	19.62
Spanish	15.11	13.93	13.06	13.93	13.19	14.10	13.26
Italian	24.57	25.97	21.77	25.85	21.70	24.06	20.29
Average	17.93	19.25	16.54	19.06	16.72	18.75	16.45

- Language-degree of a phone: how many languages a phone appears

		Language-degree			
		4	3	2	1
Language	German	18	6	8	8
	French	18	6	7	26
	Spanish	18	4	1	7
	Italian	18	5	4	6

On average, both JoinAP-Nonlinear and JoinAP-Linear perform better than Flat-Phone, and JoinAP-Nonlinear is the strongest.

Experiments

- Crosslingual experiments

- Polish:

#Finetune	Flat-Phone	JoinAP-Linear	JoinAP-Nonlinear
0	33.15	35.73	31.80
10 minutes	8.70	7.50	8.10

- Mandarin:

#Finetune	Flat-Phone	JoinAP-Linear	JoinAP-Nonlinear
0	97.10	89.51	88.41
1 hour	25.39	25.21	24.86

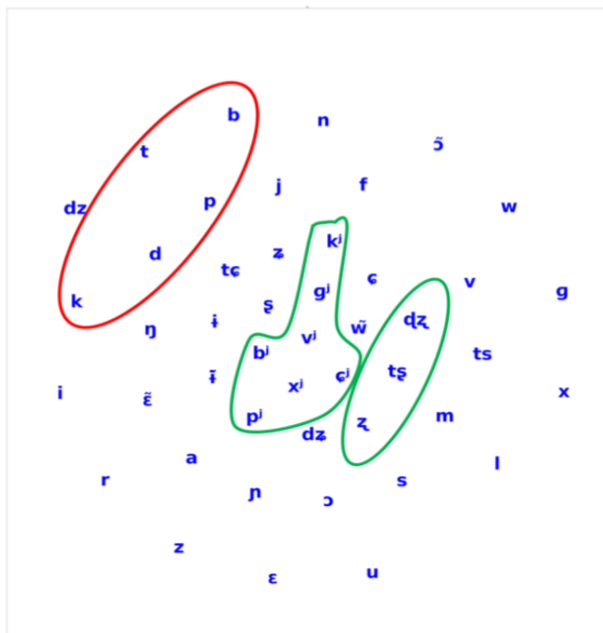
- Statistics about Polish and Mandarin:

Language	#Phones	#Unseen phones
Polish	46	18
Mandarin	96	79

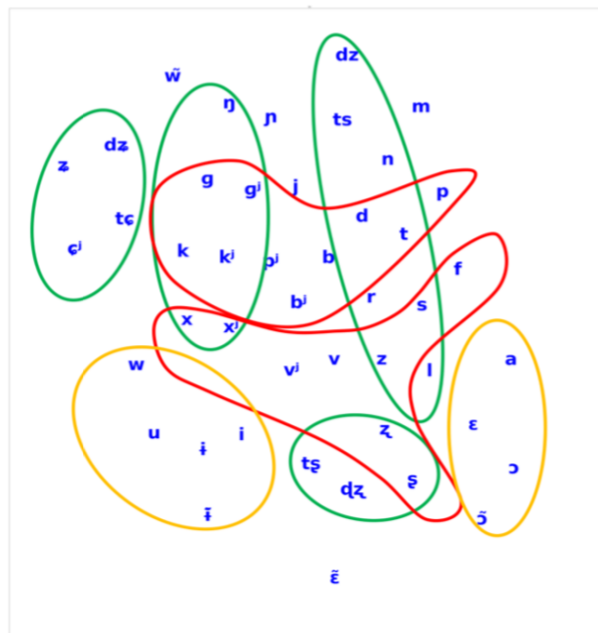
On average, both JoinAP-Nonlinear and JoinAP-Linear perform better than Flat-Phone, and JoinAP-Nonlinear is the strongest.

Experiments

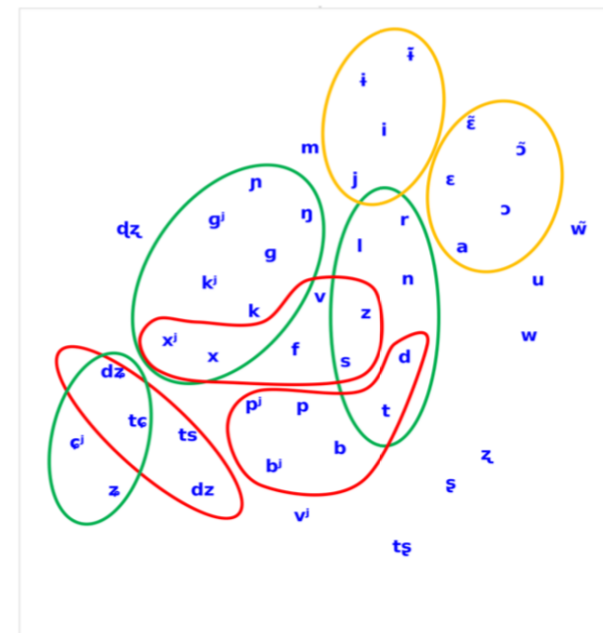
- t-SNE map of Polish phone embeddings
(obtained from un-finetuned multilingual models)



(a)



(b)



(c)

(a) Flat phone embeddings, (b) JoinAP-Linear phone embeddings, (c) JoinAP- Nonlinear phone embeddings.

Consonants with the same manner of articulation

Consonants with the same place of articulation

Vowel with similar height

Section Conclusion

- In the multilingual and crosslingual experiments, **JoinAP-Nonlinear** generally performs better than **JoinAP-Linear** and the traditional **flat-phone** method on average. The improvements for target language depend on its data amount and language-degree.
- Our JoinAP method provides **a principled, data-efficient approach** to multilingual and crosslingual speech recognition.
- Promising directions: exploring DNN based phonological transformation, and pretraining over increasing number of languages.

提纲

一、语音识别简史与基础

二、端到端语音识别

三、数据高效

四、多语言与跨语言语音识别

五、总结及展望

“WER we are and WER we think we are”

“The conclusions are clear: we are definitely not where we think we are in terms of WERs (Word Error Rates).”

ASR	CCC	SWBD	CallHome
ASR 1	17.9	11.62	17.69
ASR 2	19.2	11.45	18.6
ASR 3	16.5	10.2	15.85

Table 1: WER [%] comparison on benchmarks

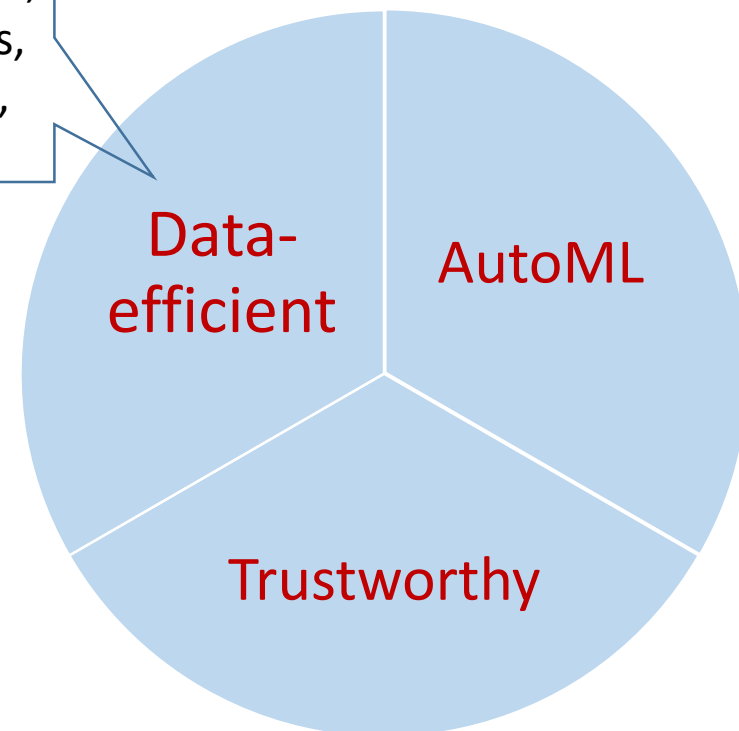
- Test: three different state-of-the-art commercial ASR solutions
- Call Center Conversations (CCC)
- The commercial ASR systems in our evaluation achieve **nearly double** the error rates (reported in the literatures) on both HUB’05 evaluation subsets.

Summary

新一代语音识别技术的若干特点

✓ Data-efficient, AutoML, Trustworthy

noises,
accents,
languages,
scenarios,
domains,
...



数据高效的多语言与跨语言语音识别

▶ CTC-CRF: 支持分立的AM与LM

- 在原理上克服了历史上各类序列鉴别模型的不足!
- 减少对大量人工标注语音数据的依赖

▶ JointAP: 联合声学与音韵学

- 促进多语言训练时信息共享以及跨语言语音识别时信息迁移



Thanks for your attention!

