WeNet 开源社区

社区主页 https://wenet.org.cn/

张彬彬

2022/10/27

CONTENTS

/01

WeNet 开源社区

社区主页 https://wenet.org.cn/

社区简介

- 社区主页：https://wenet.org.cn/
- WeNet 开源社区是什么?
  - 开源语音社区，也是国内最大的开源语音社区。
  - We 寓意“共创共赢”。
- 社区 Slogan：**让 AI 变得更简单**
- 社区的目标
  - 推动基于深度学习的语音技术落地。
  - 推动开源语音生态建设。
  - 助力国产平台和芯片生态体系。
- 社区公众号：

社区生态

- 社区生态建设
  - 行业影响力：方案、数据在行业内广泛应用。
  - 知识渠道：官方公众号、8+ 微信交流群、知乎、语音之家 WeNet 专区。
  - 社区合作：语音杂谈、语音之家、深蓝学院、SpeechIO 等。
- 社区 Sponsor：高校、数据公司、企业、个人开发者等。

Sponsors

社区项目行业落地

华为	腾讯
京东	网易
虎牙	58 同城
喜马拉雅	作业帮
蔚来	小鹏
理想	商汤
Nvidia	昆仑芯
寒武纪	地平线

58同城：WeNet端到端语音识别大规模落地方案

年处理1000万小时录音文件，支持5000万次语音对话。

虎牙在 WeNet 中开源 ONNX 推理支持

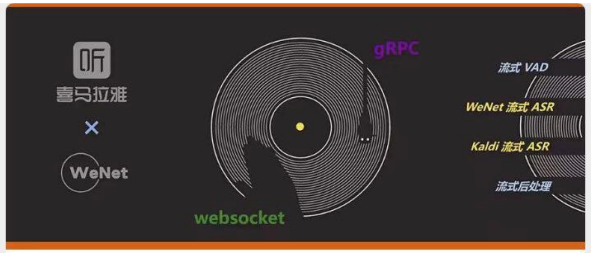
虎牙在 WeNet 中开源 ONNX 推理支持，相对 Libtorch 相对提速 20%。

NVIDIA Triton 助力网易互娱 AI Lab，改善语音识别效率及成本

网易互娱 AI Lab 的研发人员，基于 Wenet 语音识别工具进行优化和创新，利用 NVIDIA Triton 推理服务器的 GPU Batch Inference 机制加速了语音识别的速度，并且降低了成本。

京东：基于 WeNet 的端到端语音识别优化方案与落地

京东科技把 Wenet 的技术方案落地到了京东内部的 IM 沟通工具中的语音识别上，最终 Wenet 技术方案比原有线上 kaldi 的系统在京东内部咚咚 IM 中 CER 相对下降 50%，字准率提升到 90% 以上。



喜马拉雅：基于 WeNet 和 gRPC 的语音识别微服务架构的设计和应用

近日，喜马拉雅语音团队在 wenet 中增加了基于 gRPC 的流式语音识别的支持。本文由喜马拉雅语音团队撰写，介绍

作业帮：基于 WeNet + ONNX 的端到端语音识别方案

作业帮基于 WeNet 在非常短的时间内搭起一套完整的语音识别系统，并且基于 WeNet 的 U2 模型，在很多场景下都能获得非常不错的效果。

社区项目

5000+ Stars，200+ Contributors

	项目	功能
算法	**wenet**	端到端语音识别
	wetts	端到端语音合成
	wekws	端到端唤醒
	wespeaker	端到端说话人项目
	WeTextProcessing	新一代文本正规化/反正规化工具
数据	**WenetSpeech**	一万小时大规模多领域中文语音开源数据集
	Opencpop	首个开源中文歌唱合成数据集

WeNet

/02

端到端语音识别项目 WeNet 最新进展

https://github.com/wenet-e2e/wenet/

WeNet 关于流式的思考

- WeNet 流式方案：**多快好省**

多

好

WeNet: joint CTC/Attention

VS

RNN-T
Google/Microsoft/Facebook/k2

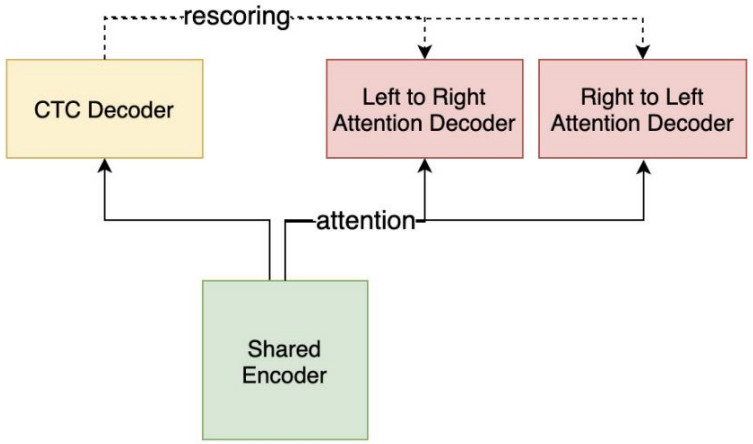
快

省

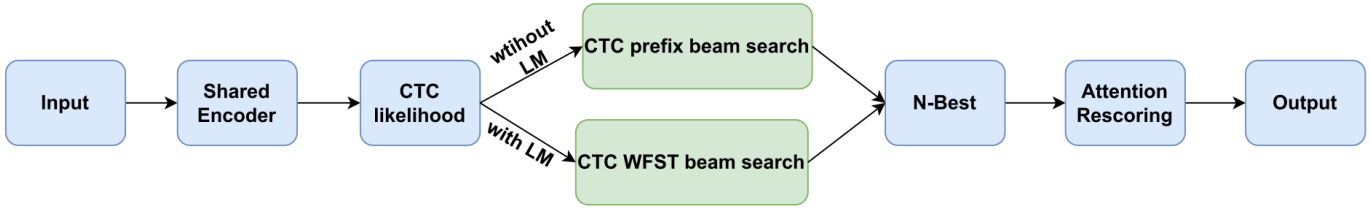
WeNet 更新：2.0 四大更新

WeNet 2.0: More Productive End-to-End Speech Recognition Toolki
https://arxiv.org/pdf/2203.15455.pdf

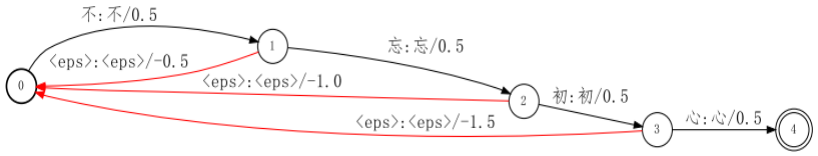
1）U2++ 算法



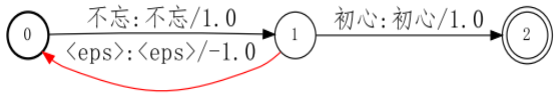
2）统一语言模型支持



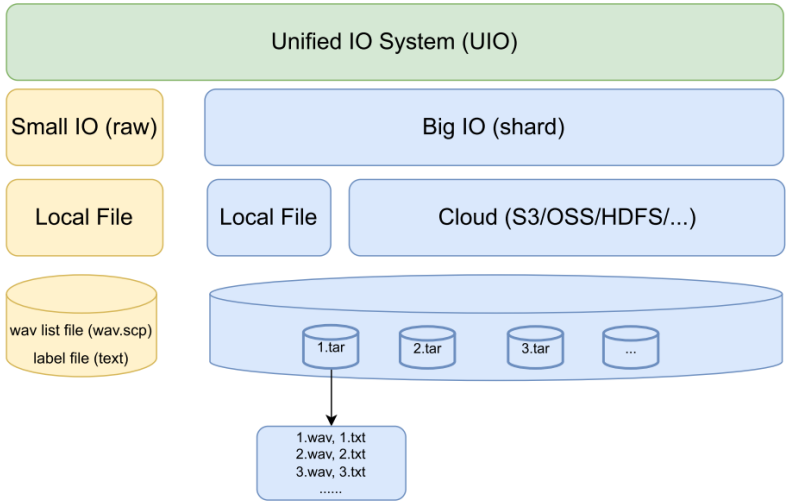
3）工业级热词方案



(a) Char level context graph without LM



4）超大规模数据训练支持



WeNet 更新：流式再思考，如何降低延迟？TrimTail

TrimTail: Low-Latency Streaming ASR with Simple but Effective Spectrogram-Level Length Penalty https://arxiv.org/pdf/2211.00522.pdf

- 问题：如何降低流式语音识别中的 Token 预测延迟?
- 现有方案如何解决?
  - Constrained Alignments
  - 损失函数改进
    - Google FastEmit
    - K2 Delayed penalized transducer

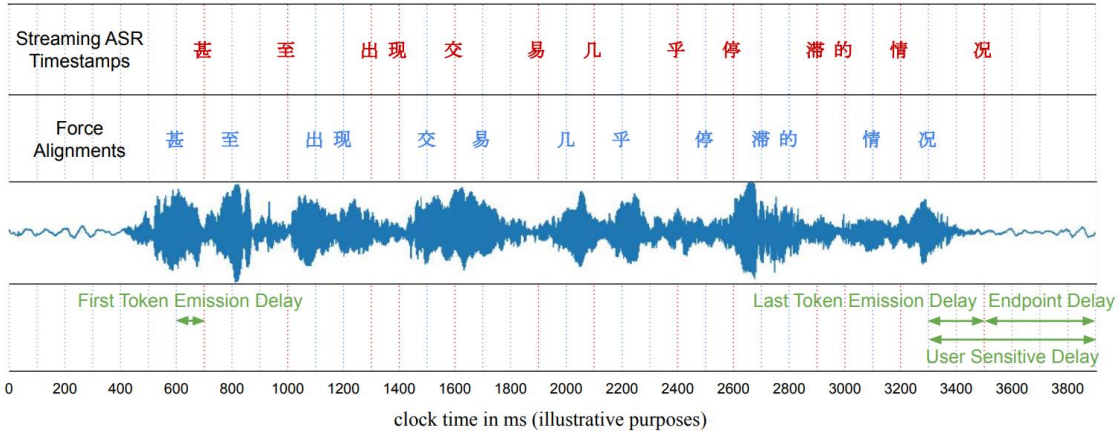
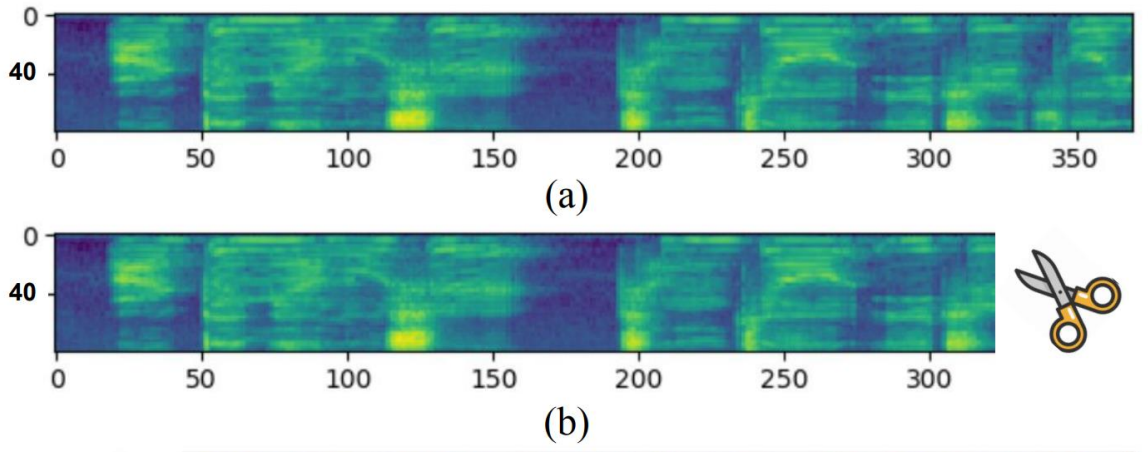


Fig. 3. A visual illustration of timeline and latency metrics of a streaming ASR system.

- 我们的方案 TrimTail，如右图所示
- TrimTail 优点
  - 无需先验对齐：相比 Constrained Alignments 方案，无需对齐。
  - 普适性： 对各种模型结构、各种损失函数（CTC/RNN-T）均适用。
  - 简单高效，即插即用：类 SpecAug，实现简单，部分测试中甚至带来了更好的识别结果。



WeNet 更新：流式再思考，如何降低延迟？TrimTail

TrimTail: Low-Latency Streaming ASR with Simple but Effective Spectrogram-Level Length Penalty https://arxiv.org/pdf/2211.00522.pdf

Models	WER (%)	FTD50 (ms)	FTD90 (ms)	LTD50 (ms)	LTD90 (ms)	AvgTD50 (ms)	AvgTD90 (ms)
Aishell-1 (test) 1/4 subsample							
Transformer (CTC)	6.92 / 5.63	70	120	30	90	68	92
+*TrimTail(50)*	**6.78** / 5.65	90 (↑20)	140 (↑20)	30 (∼)	110 (↑20)	101 (↑33)	142 (↑50)
+*TrimTail(70)*	6.83 / **5.61**	60 (↓10)	120 (∼)	-50 (↓80)	30 (↓60)	40 (↓28)	68 (↓24)
+*TrimTail(100)*	7.20 / 5.75	40 (↓30)	100 (↓20)	-100 (↓130)	-50 (↓140)	-34 (↓102)	-13 (↓ 105)
Conformer (CTC)	**5.81 / 5.05**	230	280	210	250	235	260
+*TrimTail(50)*	5.83 / 5.09	220 (↓10)	260 (↓20)	60 (↓150)	120 (↓130)	188 (↓47)	210 (↓50)
+*TrimTail(70)*	6.03 / 5.17	150 (↓80)	200 (↓80)	-10 (↓220)	50 (↓200)	109 (↓126)	132 (↓128)
+*TrimTail(100)*	6.48 / 5.41	100 (↓130)	140 (↓140)	-80 (↓290)	-20 (↓270)	26 (↓209)	52 (↓208)
Conformer (Transducer)	6.96 / N/A	290	350	200	230	270	284
+*TrimTail(50)*	7.09 / N/A	190 (↓100)	250 (↓100)	40 (↓160)	80 (↓150)	156 (↓114)	175 (↓109)
+*TrimTail(70)*	**6.94** / N/A	150 (↓140)	210 (↓140)	-30 (↓230)	20 (↓210)	98 (↓172)	121 (↓163)
+*TrimTail(100)*	7.36 / N/A	130 (↓160)	190 (↓160)	-60 (↓260)	-10 (↓240)	58 (↓212)	80 (↓204)
Conformer (CTC + Transducer)	**6.34/5.69**	250	310	140	190	234	247
+*TrimTail(50)*	6.47/5.78	180 (↓70)	220 (↓90)	10 (↓130)	60 (↓130)	138 (↓96)	155 (↓92)
+*TrimTail(70)*	6.71/5.87	120 (↓130)	180 (↓130)	-80 (↓220)	-20 (↓210)	56 (↓178)	73 (↓174)
+*TrimTail(100)*	7.28/6.16	100 (↓150)	170 (↓140)	-120 (↓260)	-90 (↓280)	5 (↓229)	28 (↓219)
Aishell-1 (test) 1/8 subsample							
Conformer (CTC)	5.85 / 5.16	150	200	110	170	148	175
+*TrimTail(50)*	**5.83 / 5.16**	80 (↓70)	130 (↓70)	-30 (↓140)	40 (↓130)	54 (↓94)	80 (↓95)
+*TrimTail(70)*	5.96 / 5.24	50 (↓100)	110 (↓90)	-110 (↓220)	-30 (↓200)	-13 (↓161)	16 (↓159)
+*TrimTail(100)*	6.54 / 5.52	20 (↓130)	90 (↓110)	-180 (↓290)	-110 (↓280)	-93 (↓241)	-60 (↓235)
Librispeech (test_clean) 1/4 subsample							
Conformer (CTC)	4.84 / 4.13	-	-	-	-	-	-
+*TrimTail(50)*	**4.68 / 4.01**	- (↓80)	- (↓40)	- (↓80)	- (↓40)	- (↓87)	- (↓70)
+*TrimTail(70)*	4.69 / 4.02	- (↓120)	- (↓80)	- (↓120)	- (↓80)	- (↓124)	- (↓104)
+*TrimTail(100)*	4.82 / 4.12	- (↓160)	- (↓80)	- (↓120)	- (↓80)	- (↓143)	- (↓123)
+*TrimTail(150)*	5.18 / 4.31	- (↓160)	- (↓120)	- (↓160)	- (↓80)	- (↓155)	- (↓135)
+*TrimTail(200)*	5.24 / 4.38	- (↓160)	- (↓120)	- (↓160)	- (↓120)	- (↓167)	- (↓144)

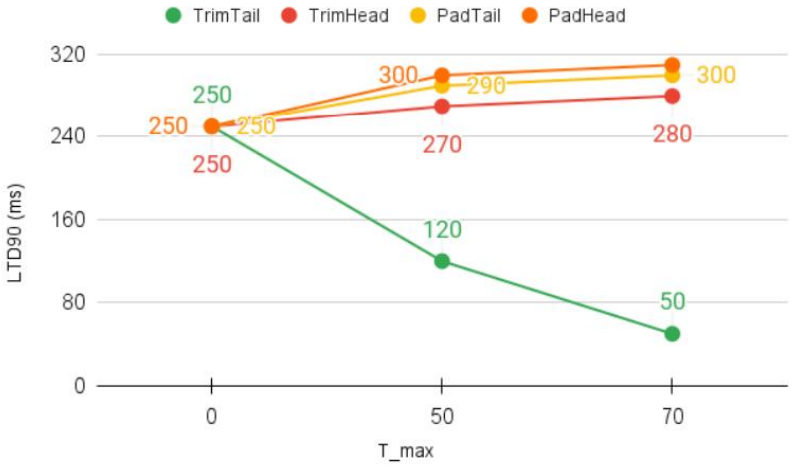
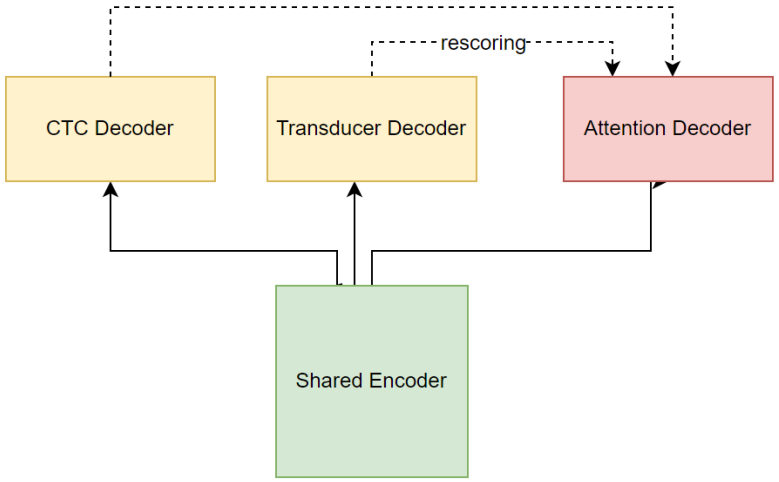


Table 2. WER comparison of different cutting lengths .

Cutting Length (ms)	Baseline	Baseline (+*TrimTail(50)*)
0	**5.81 / 5.05**	5.83 / 5.09
200	5.93 / 5.16	**5.83 / 5.09**
300	7.26 / 5.89	**5.85 / 5.09**
400	10.33 / 8.12	**6.0 / 5.16**
500	13.08 / 10.95	**7.79 / 6.03**

WeNet 更新：支持 RNN-T

详情链接：https://mp.weixin.qq.com/s/7tPhyQQ9saITobz3Hs7bqA



Joint CTC & Transducer & Attention

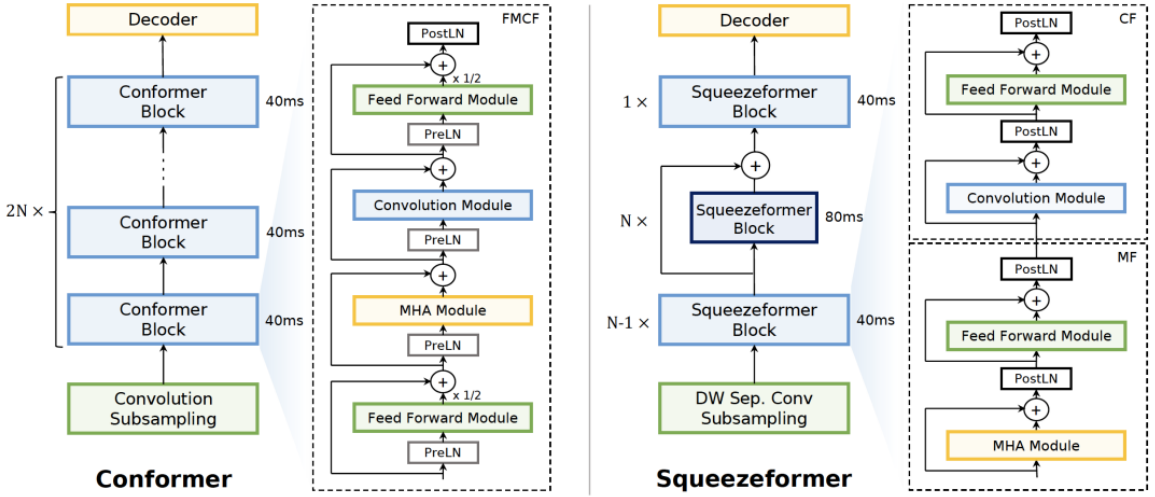
训练损失函数			解码方式	
RNN-T	CTC	Attention	greedy	rescoring
✗	✓	✓	4.94	4.61
✓	✗	✗	5.64	/
✓	✗	✓	5.03	4.87
✓	✓	✓	**4.88**	**4.45**

AIShell-1 非流式实验结果

实验详情：https://github.com/wenet-e2e/wenet/tree/main/examples/aishell/rnnt

WeNet 更新：支持 Squeezeformer

详情链接：
https://mp.weixin.qq.com/s/AuQD5Hwd37D52vENlimRIw



实验1：中等模型对比

Model	FLOPs	Params	Test clean		Test other	
			ctc greedy	attn rescore	ctc greedy	attn rescore
Conformer-M	31.7G	33.2M	3.51	3.18	9.57	8.72
Squeezeformer-SM12-V0	19.7G	21.2M	3.51	3.07	9.28	8.44
Squeezeformer-SM12-V1	26.3G	33.2M	3.47	3.10	8.85	8.03
Squeezeformer-SM12-V2	31.8G	34.0M	**3.30**	**2.96**	**8.58**	**7.91**

实验2：大模型对比

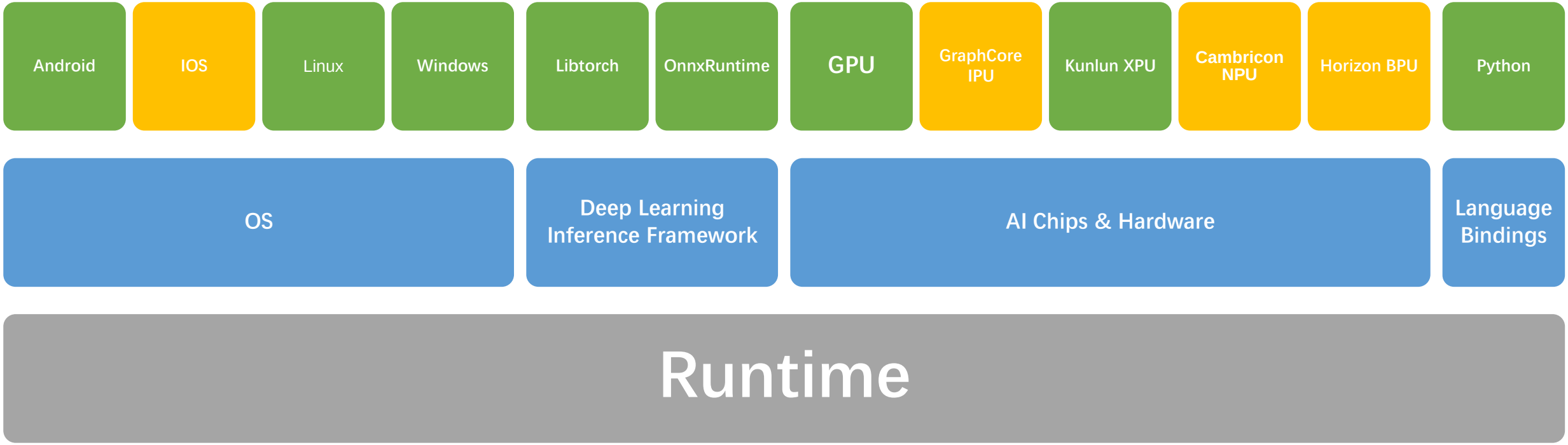
Model	FLOPs	Params	Test clean		Test other	
			ctc beam	attn rescore	ctc beam	attn rescore
Conformer-L	89.6G	81.7M	2.96	2.66	7.14	6.53
Squeezeformer-ML12	76.6G	81.7M	**2.72**	**2.45**	**6.52**	**5.85**

实验3：流式模型对比

Model	FLOPs	Params	Chunk	Test clean		Test other	
				ctc beam	attn rescore	ctc beam	attn rescore
Conformer-U2++	31.7G	33.2M	full	3.76	3.32	9.50	8.67
			16	4.54	3.80	11.52	10.38
Squeezeformer-U2++	26.3G	33.2M	full	**3.45**	**3.07**	**8.29**	**7.58**
			16	**4.34**	**3.71**	**10.60**	**9.60**

WeNet 更新：全新 Runtime 设计

- 落地 Runtime
  - 多种系统支持
  - 多种深度学习推理框架支持
  - 多种 AI 芯片支持
  - 多语言支持



WeNet 更新：3.0 Roadmap https://github.com/wenet-e2e/wenet/blob/main/ROADMAP.md

WeNet Roadmap

This roadmap for WeNet. WeNet is a community-driven project and we love your feedback and proposals on where we should be heading.

Please open up issues or discussion on github to write your proposal. Feel free to volunteer yourself if you are interested in trying out some items(they do not have to be on the list).

WeNet 3.0 (2023.06)

- [x] ONNX support
- [x] RNN-T support
- [ ] Vosk like models and API for developers.
  - [ ] Models(Chinese/English/Japanese/Korean/French/German/Spanish/Portuguese)
  - [ ] API(python/c/c++/go/java)
- [ ] Self training, streaming
- [ ] Light weight, low latency, on-device model exploration
- [ ] Audio-Visual speech recognition
- [ ] Platforms
  - [x] Raspberry Pi
  - [ ] Harmony OS
- [ ] ASIC XPU
  - [ ] Horizon Journey
  - [ ] GraphCore
  - [ ] TO ADD

/03

社区新发布项目 WeKws/WeSpeaker/WeTextProcessing 简介

端到端唤醒项目 WeKws

WeKws: A production first small-footprint end-to-end Keyword Spotting Toolkit
https://arxiv.org/pdf/2210.16743.pdf

- 项目动机
  - 工业界唤醒方案众多，且部分方案训练、部署难度大。
  - 部署的芯片和平台众多，适配工作量大。
  - 业界缺乏一个好用的、统一的、针对语音唤醒任务的开源的框架。
- WeKws 项目特点
  - 产品优先：流式支持，多种 Runtime 支持
  - 端到端：完全端到端方案
  - 轻量级：
  - 高准确率：开源数据集取得领先效果

Table 1. FRR(%) comparison of the proposed WeKws and other KWS systems with FAH fixed at 0.5, on keywords "Hi xiaowen" and "Nihao wenwen".

System	#Params	Hi xiaowen	Nihao wenwen
Wang at al. [10]	57k	0.7	0.5
WeKws	153k	**0.50**	**0.43**

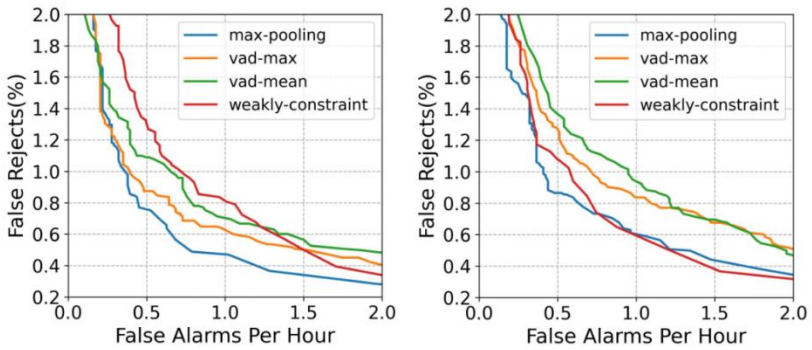
Table 2. FRR (%) comparison of the proposed WeKws and other KWS systems on keyword "Hey snips".

System	#Params	FAH=0.5	FAH=1
Zhang at al. [5]	244k	3.53	2.82
Coucke at al. [20]	222k	**0.12**	–
WeKws	153k	**0.12**	**0.08**

Table 3. Comparison of the proposed WeKws and other KWS systems on Google Speech Command.

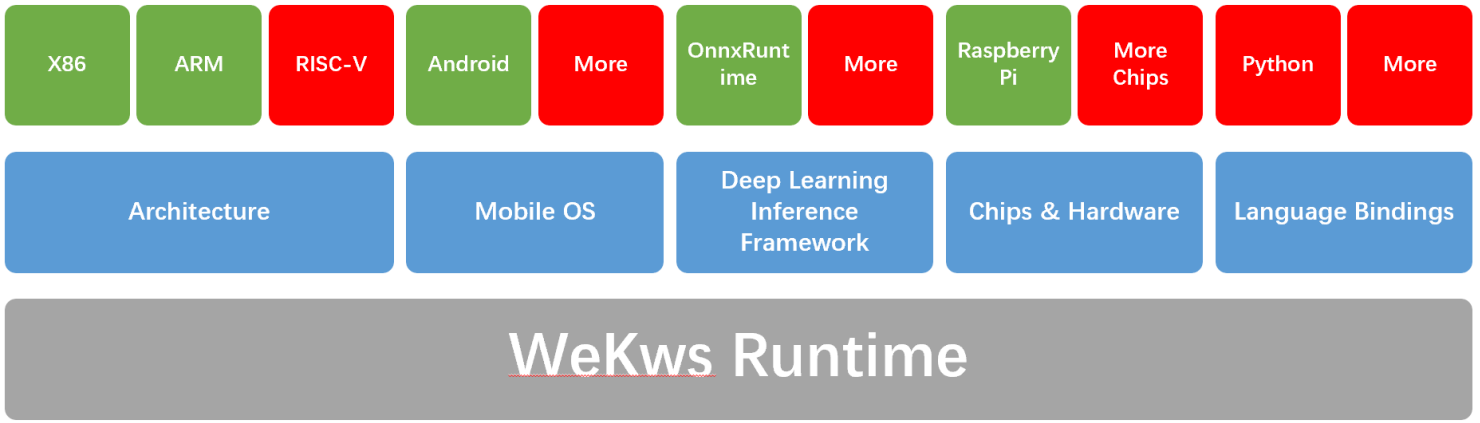
System	#Params	Accuracy (%)
Zhang at al. [21]	109K	97.20
Ding at al. [22]	404K	97.56
WeKws	158K	**97.97**

1) MobvoiHotwords，Hey Snips, Google Speech Command 三大开源数据集取得领先实验结果



(a) Hi xiaowen (b) Nihao wenwen

2) 端到端训练



3) 产品优先的 Runtime 支持

端到端说话人项目 WeSpeaker

Wespeaker: A Research and Production oriented Speaker Embedding Learning ToolkitToolkit https://arxiv.org/pdf/2210.17016.pdf

- WeSpeaker 项目特点
  - **高质量**：在开源数据集说话人确认、说话人聚类上取得较有竞争力的效果。
  - **轻量级**：聚焦于说话人表征学习任务
  - **支持超大规模数据**：支持从几小时到数万小时的工业级数据的模型训练
  - **在线数据增强**：在线重采样、加噪、加混响、速率扰动
  - **提供部署方案**：OnnxRuntime，GPU Triton 生产环境支持

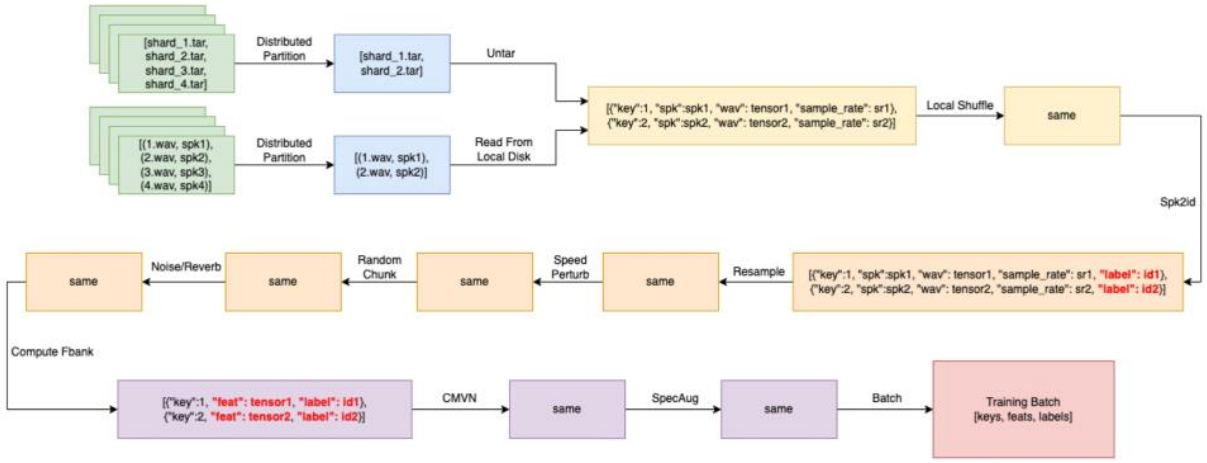


Fig. 3. The pipeline of online feature preparation in Wespeaker. "same" means that no new attributes are added and only change a specific attribute value for each sample (except local shuffle), compared with the last node.

Table 1. Results achieved using different architectures on the VoxCeleb dataset, "dev" of part 2 is used as the training set

Architecture	voxceleb1_O		voxceleb1_E		voxceleb1_H	
	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
TDNN	1.590	0.166	1.641	0.170	2.726	0.248
ECAPA-TDNN ([4])	0.870	0.107	1.120	0.132	2.120	0.210
ECAPA-TDNN	0.728	0.099	0.929	0.100	1.721	0.169
ResNet34 ([3])	1.31	0.154	1.38	0.163	2.50	0.233
ResNet34	0.723	0.069	0.867	0.097	1.532	0.146
ResNet221	0.505	0.045	0.676	0.067	1.213	0.111
ResNet293	**0.447**	**0.043**	**0.657**	**0.066**	**1.183**	**0.111**

Table 2. Results on the CNCeleb evaluation set

Architecture	EER(%)	minDCF
TDNN	8.960	0.446
ECAPA-TDNN	7.395	0.372
ResNet34([16])	9.141	0.463
ResNet34	6.492	0.354
ResNet221	**5.655**	**0.330**

Table 3. Results on the VoxConverse dev set

System	MISS(%)	FA(%)	SC(%)	DER(%)
[33] (system SAD)	2.4	2.3	3.0	7.7
Wespeaker (system SAD)	4.4	0.6	2.1	7.1
Wespeaker (oracle SAD)	2.3	0.0	1.9	4.2

新一代 TN/ITN 工具 WeTextProcessing

发布链接：
https://mp.weixin.qq.com/s/q_11lck78qcjylHCi6wVsQ

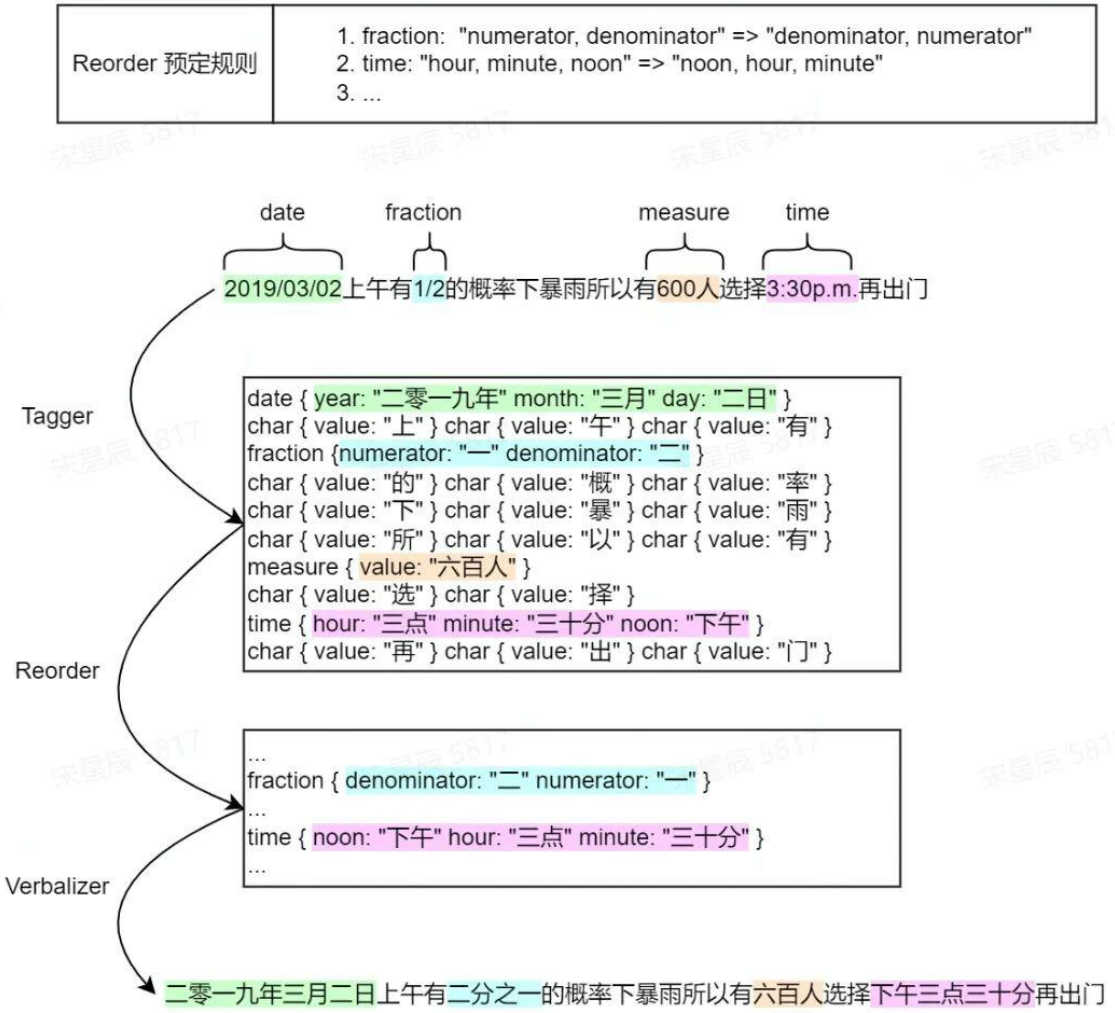
- WeTextProcessing 项目特点
  - 产品优先：
    - 基于语法规则的 WFST 方案，准确可控
    - 快速修复 Bug，依托 pynini，用户可以 python 语言修复 badcase。
  - 简单易用：
    - python 环境一键 pip 安装
    - 生产环境仅依赖 OpenFST。

一键安装，开箱即用

```
# install
pip install WeTextProcessing
```

```
# tn usage
>>> from tn.chinese.normalizer import Normalizer
>>> normalizer = Normalizer()
>>> normalizer.normalize("2.5平方电线")
# itn usage
>>> from itn.chinese.inverse_normalizer import InverseNormalizer
>>> invnormalizer = InverseNormalizer()
>>> invnormalizer.normalize("二点五平方电线")
```

TN 算法流程



/04

社区数据集 Opencpop 和 WenetSpeech 介绍

社区数据集——1万小时开源数据集 WenetSpeech

- 数据量：10000+ 小时高质量标注数据；2400+ 小时弱监督标注数据；22400+ 小时总音频
- 多领域：囊括有声书、解说、纪录片、电视剧、访谈、新闻、朗读、演讲、综艺和其他等10大场景
- 2个人工精标测试集：TEST_NET，TEST_MEETING

1500+ 申请，含高校、研究所、企业等

省份	流量	流量占比
北京	44.9TB	33.56%
广东	24.98TB	18.67%
上海	16.69TB	12.48%
江苏	10.15TB	7.59%
浙江	9.33TB	6.97%
四川	5.03TB	3.76%

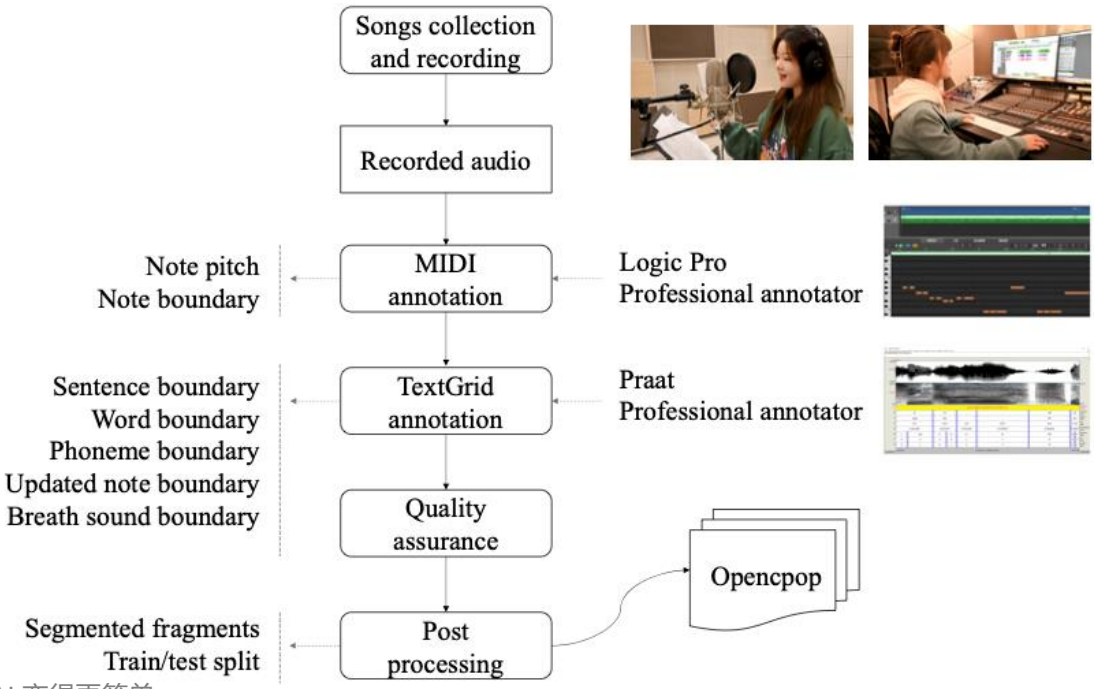
共 32 条　　1 / 6 页

注：WenetSpeech 由腾讯天籁实验室提供的下载服务

社区数据集——首个中文歌唱合成数据集 Opencpop

- 数据概况：100 首；MIDI/TextGrid 标注；全音素覆盖
- 官网：https://wenet.org.cn/opencpop/
- **600+ 申请，含高校、研究所、企业等**

合成样例，攒钱回家过大年



什么是开源?

崔宝秋

小米集团副总裁

WeNet

欢迎各位有志同学投身开源

一起 让 AI 变得更简单！