语音识别技术基础

罗平峰

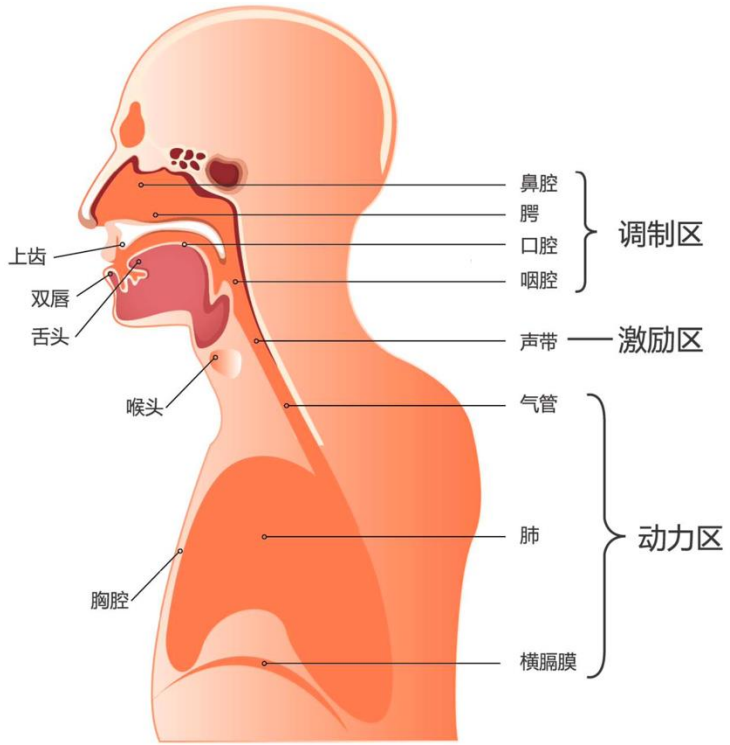
2023

目 录

语音信号的产生

激励响应模型： 人脑根据要表达的信息，控制肺部产生气流，经过气管引起声带振动形成声源(通常称为激励)；最后经过声道(咽腔、口腔、鼻腔等区域)调制，产生了我们所听到的语音。

信号的采集： 麦克风将声波转换成电压（电磁感应），完成模数信号转换，方便计算机处理。



语音识别的原理

基本问题： 给定一系列观测信号，在语言空间中找到最可能的文本序列。

数学原理（后验概率最大）：

$$W^* = \text{argmax}_W \ P(W|X)$$

若使用贝叶斯定理，则可得

$$P(W|X) = \frac{P(X|W)P(W)}{P(X)}$$

$$\propto P(X|W)P(W)$$

方法一： 拆分建模的思路为传统的语音识别方法，即**声学模型P（X | W）**、**语言模型P(W)**独立建模。有GMM-HMM、DNN-HMM等，其中Kaldi、HTK等工具中实现主要就是这两种。

方法二： 直接对**P(W | X)**进行建模，即声学和语言模型放在一个系统进行联合建模，则为目前的端到端的语音识别方法。有CTC、RNNT、AED等，其中espnet、K2等工具实现的是这类最新的E2E方法。

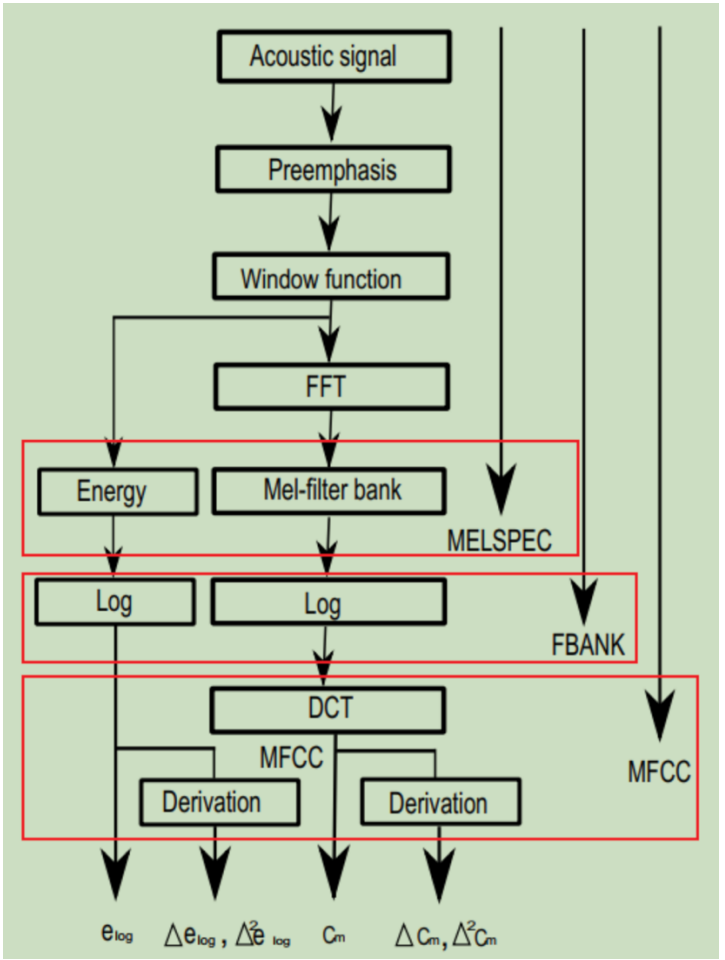
语音识别的方法——信号处理(原始信号复杂多变)

特征提取

- 分帧：一般取**25ms**（太长则不满足短时平稳假设，太短则无法表征特征）
- 预加重：缓解高频能量的衰减
- 加窗：缓解频谱泄漏的现象
- FFT：时频转换（三角函数的正交性）
- 三角滤波：仿人耳声学感知变换和减少参数量（每个bin合并成一个能量点计算）
- 取对数：非线性变换

问题

- 传统傅立叶分析的局限性？（平稳假设、相同频率分辨率、时间分辨率为零）
- 联合时频分析方法：短时傅里叶变换(上述特征提取部分)、小波分析等
- 直接从raw signal建模 （类似图像，RGB信号直接输入模型）



语音识别的方法——信号处理

前端处理（改善信号质量）

- 加性噪声
  - 谱减法，假设噪声和原始语音的能量谱叠加得到带噪信号，估计出噪声能量谱，相减和平滑即可。
- 混响和回声
  - 估计出声源到接收端的传递函数（房间的耐冲响应函数表示RIR），设计滤波器和RIR抵消
  - 设计逆滤波器使得生成的LPC参数非高斯化
  - 基于T60(衰减60db)估计RIR，然后利用谱减法
  - 线性预测模型（当前的信号由历史的信号延迟衰减并叠加当前信号形成）
- 信道差异：覆盖和补偿
- 麦克风阵列
  - 阵列类型：线性阵列、环形阵列
  - 增益是入射角的函数；控制每路麦克风的延时即可控制指向性（相位一致）
  - 不同麦克风接受的噪音不相关，叠加则会抵消
  - 可利用时间和空间信息，实现方向选择(延迟加和)、去噪、去混响
- CMVN：可以理解为一种高通滤波器（滤掉固定不变的成分）
- DAE：干净数据和带噪数据，去噪自编码器

语音识别的方法——传统语音识别

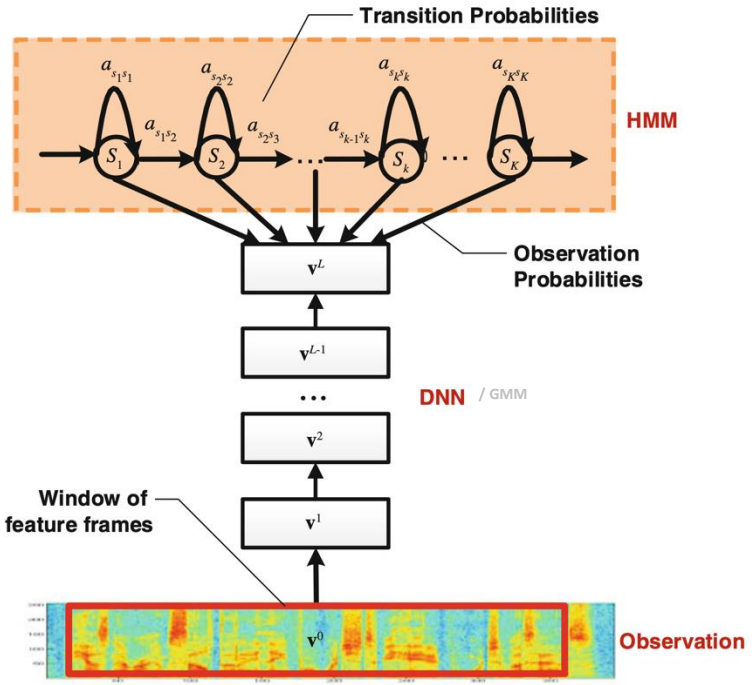
根据前述，是否对后验概率直接建模，语音识别可以分成E2E和传统方法。

传统方法的思路：通过声学模型将信号识别成音素序列，音素序列在声学和语言模型的共同约束下识别成字词序列。

P(X | W) 建模（声学模型）

- GMM-HMM
  - 建模单元：一般选择音素，考虑到音素上下文相关和协同发音等信息，会进一步使用三音素或者双音素，并通过聚类（合并相近的类，减少数量）得到最终的建模单元。
  - 模型结构：每个建模单元（三音素或双音素）都用一个HMM表示，包含转移概率和发射概率(GMM模型建模)，后者一般更重要而前者可以取固定的值。
  - 参数估计：GMM和HMM的参数使用EM算法
    - E步：
      - 根据现有参数计算$P(ot \mid m)$
    - M步：
      - 根据$P(ot \mid m)$ 更新GMM参数
    - 不断重复E和M步，直到收敛
- DNN-HMM
  - 建模单元：和GMM-HMM相同，为三音素或者双音素
  - 模型结构：
    - 将GMM-HMM中DNN替换成DNN
    - DNN可以是TDNN、LSTM、CNN等网络结构
  - 参数估计：基于LF-MMI的loss和梯度下降

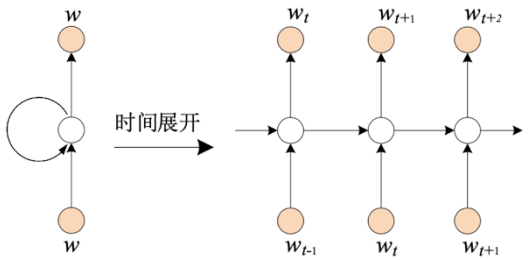
$$\mathcal{F}_{MMI} = \sum_u \log \frac{p(\mathbf{O}_u|S_u)^\kappa P(W_u)}{\sum_W p(\mathbf{O}_u|S)^\kappa P(W)}$$



语音识别的方法——传统语音识别

P(W)建模（语言模型）

- **Ngram**
  - 使用N阶的马科夫模型建模：给定历史词汇的条件下当前词汇的概率
  - 通过统计词频和利用平滑算法估计文法的概率值
  - 公式：$P(wi \mid wi-n+1,wi-n+2,...,wi-1) = count(wi-n+1,wi-n+2,...,wi) \ / \ count\ (wi-n+1\ ,\ wi-n+2\ ,\ ...,\ wi-1\ )$
- **NNLM模型**
  - 使用神经网络建模：给定其它词汇的条件下预测当前词汇的概率
  - 网络输出单元的集合等于词典，基于梯度下降学习



- **Ngram和NNLM对比**

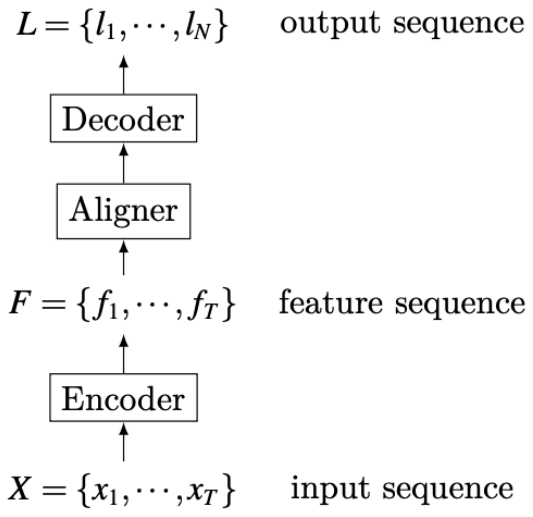
算法	优势	劣势
Ngram	• 存储是文法的概率值，访问复杂度O(1) • 方便编辑，例如领域适应、文法概率惩罚或激励（如热词应用）	• 建模长历史信息的能力较弱(N阶假设约束) • 阶数增加则参数呈指数倍增加
NNLM	• 建模较长的历史信息	• 训练好后参数不易修改 • 文法概率需要临时计算，耗时较大（有些优化算法可以将NNLM转成Ngram格式存储）

语音识别的方法——E2E语音识别

根据前述，是否对后验概率直接建模，语音识别可以分成E2E和传统方法。
E2E方法的思路：直接对P(W | X)进行建模，即声学和语言模型联合优化。

E2E语音识别建模方法

- E2E方法的基本结构和数学模型对比
  - Xt 是输入序列X的第t个信号，Yt\Yu是文本(标签)序列的第t\u个



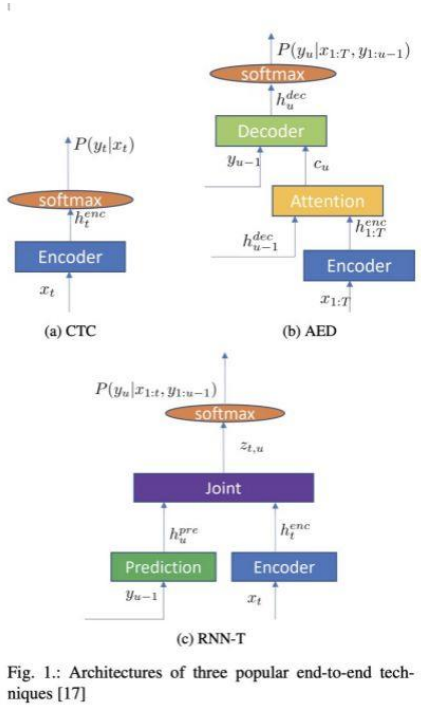


Fig. 1.: Architectures of three popular end-to-end techniques [17]

E2E	model	math
CTC	Encoder	P(Yt | Xt)
AED	Encoder/Attention/Decoder	P(Yu | X1:T, Y1:u-1)
RNNT	Transciptor/Preditor/Joiner	P(Yu | X1:t, Y1:u-1)

语音识别的方法——E2E语音识别

CTC

- 基本形式：
  - 给定观测信号Xt的条件下，关于Yt的概率（Yt之间独立，无显式LM）
  - 通过引入blank符号和考虑所有可能对齐解决对齐问题
  - 支持流式
- 数学意义：P(Yt | Xt)
- 公式展开：

$$P_{\text{CTC}}(\mathbf{y}|\mathbf{x}) = \sum_{\mathbf{a}\in\mathcal{B}^{-1}(\mathbf{y})} P(\mathbf{a}|\mathbf{h})$$

$$= \sum_{\mathbf{a}\in\mathcal{B}^{-1}(\mathbf{y})} \prod_{t=1}^{T'} P(a_t|h_t)$$

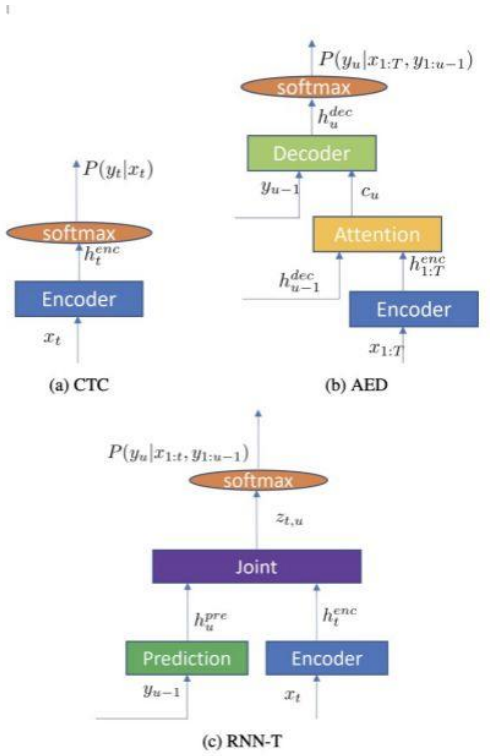


Fig. 1.: Architectures of three popular end-to-end techniques [17]

语音识别的方法——E2E语音识别

AED

- 基本形式：
  - 给定完整观测信号X**的加**权注意力和历史输出Y1:u-1的条件下，关于当前输出Yu的概率
    - 当前的output依赖于历史的output（显示的LM建模）
  - 使用注意力机制解决对齐问题
  - 标准形式使用全局注意力（流式需要改成局部注意力）
- 数学意义：P(Yu | X1:T, Y1:u-1)
- 公式展开：

$$P_{\text{Attn}}(\mathbf{y}|\mathbf{x}) = P(\mathbf{y}|\mathbf{h}) = \prod_{u=1}^{U} P(y_u|c_u, y_{<u})$$

$$c_u = \sum_{t=1}^{T} \alpha_{u,t} h_t$$

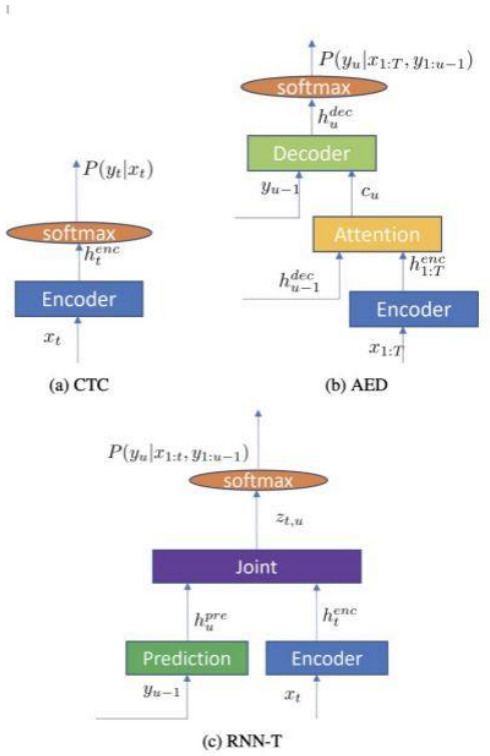


Fig. 1.: Architectures of three popular end-to-end techniques [17]

语音识别的方法——E2E语音识别

RNNT

- 基本形式：
  - 给定历史观测X1:t和历史输出Y1:u-1的条件下，关于当前输出Yu的概率
    - 当前的output依赖于历史的output（显示的LM建模）
  - 通过考虑所有可能对齐解决对齐问题
  - 支持流式
- 数学意义： P(Yu | X1:t, Y1:u-1)
- 公式展开：

$$P_{\mathrm{RT}}(\mathbf{y}|\mathbf{x}) = \sum_{\mathbf{a}\in\mathcal{B}^{-1}(\mathbf{y})} P(\mathbf{a}|\mathbf{h})$$

$$= \sum_{\mathbf{a}\in\mathcal{B}^{-1}(\mathbf{y})} \prod_{t=1}^{T'} P(a_t|h_t, y_{<u_t})$$

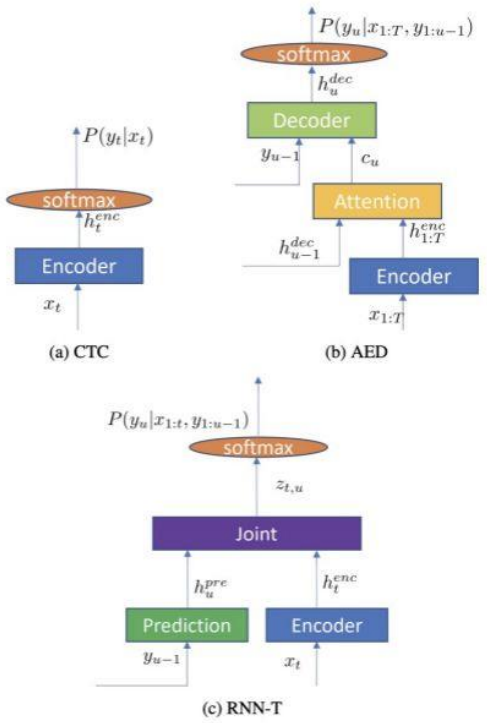


Fig. 1.: Architectures of three popular end-to-end techniques [17]

语音识别的方法——E2E语音识别

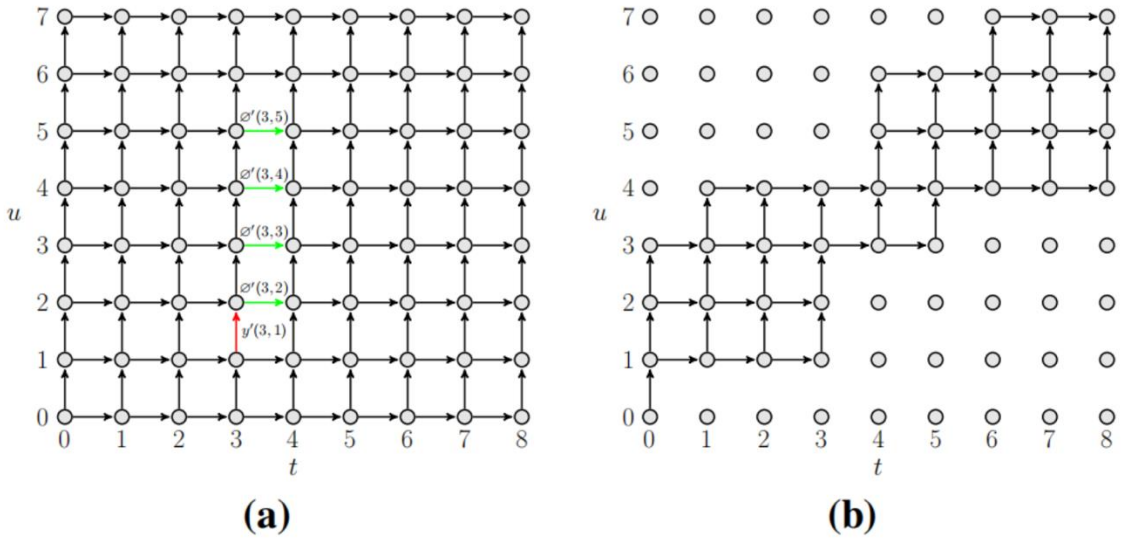
语音识别的传统方法和E2E方法对比

算法	优势	劣势
传统	模块拆分，方便独立优化特定子模块	声学和语言模型独立建模（模块建模的累计误差、系统复杂模块众多、个别模块需要专家知识）
E2E	联合建模（无累积误差、统一优化、架构简单）	纯E2E模型领域迁移效率不高 （需要重新训练，或者像传统语音识别一样通过LM做领域bias）

工业进展——K2

Pruned RNNT

- 标准的前后向算法需要考虑T和U两个维度（espnet、pytorch等工具的rnnt训练速度慢、较难实用，N*T*U*V）
- 语音和文本满足单调对齐 -> 只有对角线上小部分路径是有效的
- 先用am+lm估算有效的‘窄带’，计算am+lm+joiner只考虑窄带内的节点，使得计算复杂度从 T * U 降到 T * beam （其中beam << U）



多	更大的batch_size 更多的模型参数
快	计算复杂度降低
好	丢掉噪声干扰
省	显存占用更少

未来展望

完全端到端的语音识别	目前端到端的方法通常只是考虑声学和语言模型的同时优化(信号到文本的直接映射)； 如果能进一步，“同时优化信号处理、特征表示、声学模型、语言模型”，那么这种完全端到端方法的建模能力和鲁棒性将会更强。
低资源语音识别	方言、少数语种、特定场景的音频获取和标注都非常困难（音频数据一般成本在大几百/小时，一些敏感场景或者稀有数据甚至无法获得），如果解决低资源语音识别将极大降低算法落地成本。大致思路有：知识迁移、无监督、半监督（少量标注、海量无标注）。
鲁棒性语音识别	语音信号多变性受到信源和信道影响，例如口音、高噪声、语音混叠（**鸡尾酒会问题**）等因素严重影响语音识别效果，如何确保语音识别系统在复杂声学环境稳定的性能？大致思路：前端信号处理、后端模型增强和完全E2E。
自适应语音识别	模型根据场景、用户行为自适应，针对性地提高个体识别效果。
多模态语音识别	语音方向从单任务往多任务（多任务音频模型包括whisper、audioPaLM、SpeechPrompt等），从单模态往多模态（多模态模型包括audioPaLM、SeamlessM4T等）发展。