第三章　语音识别

第三章　语音识别

- 3.1 语音识别概述
- 3.2 声学模型
- 3.3 语言模型
  - 3.3.1 语言模型的基本概念（基本概念）
  - 3.3.2 N-gram语言模型（掌握、了解）
  - 3.3.3 神经语言模型（了解）
- 3.4 语音识别解码算法
- 3.5 语音识别技术的展望

语言模型的定义

- 语言模型：用来计算一个句子概率的模型
  - 从词表中任意选择若干同音词所构成的序列不一定能构成自然语言中的合乎句法的句子。
  - 人类在识别和理解语句时充分利用了这种约束。

- 在语音识别中的应用
  - 在大词汇量的语音识别系统中，统计语言模型被广泛应用来实现句子的约束。将这些知识与声学模型匹配相结合进行结果判决，以减小由于声学模型不够合理而产生的误识。
  - 例子：“棠园餐厅在哪里”这句话，只有声学模型可能会被误识别为“堂源餐厅在哪里”

语言模型的分类

- 基于文法的语言模型
  - 总结语法规则及语音规则
- 基于统计的语言模型
  - 统计各个词的出现概率及其相互关联的条件概率
  - 经典模型：N-gram语言模型
- 基于深度学习的语言模型（神经语言模型）
  - 通过训练网络来学习词表征和单词序列的概率化表示

语言模型的评价指标（Perplexity）

- 衡量一个模型的拟合测试数据程度。
- 使用模型分配给测试集的概率。
- 估计测试语料库中的句子概率并取倒数，对句子长度进行归一化。

$$PP(W) = \sqrt[N]{\frac{1}{P(w_1 w_2 \ldots w_N)}}$$

$$P(w_1^N) = P(w_1)P(w_2|w_1)P(w_3|w_1^2)\ldots P(w_N|w_1^{N-1}) = \prod_{n=1}^{N} P(w_n|w_1^{n-1}) \qquad W = w_1^N$$

- 测量预测下一个词时的加权平均分支因子(越低越好)

第三章　语音识别

- 3.1 语音识别概述
- 3.2 声学模型
- 3.3 语言模型
  - 3.3.1 语言模型的基本概念
  - 3.3.2 N-gram语言模型
  - 3.3.3 神经语言模型
- 3.4 语音识别解码算法
- 3.5 语音识别技术的展望

N-gram语言模型（掌握）

- 单词序列

$$w_1^n = w_1 \ldots w_n$$

- 概率的链式法则

$$P(w_1^n) = P(w_1)P(w_2 \mid w_1)P(w_3 \mid w_1^2)\ldots P(w_n \mid w_1^{n-1}) = \prod_{k=1}^{n} P(w_k \mid w_1^{k-1})$$

- N-gram近似

$$P(w_1^n) = \prod_{k=1}^{n} P(w_k \mid w_{k-N+1}^{k-1})$$

假设词w_k出现的概率只与前N-1个单词有关：N-1阶马尔可夫模型

N-gram语言模型（续）（基本概念）

- 根据之前的上下文估计每个单词的概率
  - P(phone | Please turn off your cell)

- 所需的参数数量随着先前上下文的单词数量呈指数增长
- 一个N-gram 模型只需要N–1 个先前上下文的单词
  - Unigram: P(phone)
  - Bigram: P(phone | cell)
  - Trigram: P(phone | your cell)

N-gram语言模型的复杂度（困惑度）（基本概念）

- 模型训练了来自《华尔街日报》(WSJ)的3800万个单词（可重复），词汇表的大小为19979个单词（互相不重复的单词）
- 用《华尔街日报》150万单词进行评估

	Unigram	Bigram	Trigram
Perplexity	962	170	109

没有使用语言模型（0-gram）时的perplexity为多少？

N-gram语言模型的概率估计（掌握）

- 基于单词序列的相对频率，可以从原始文本中估计出N-gram条件概率。

Bigram: $$P(w_n \mid w_{n-1}) = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

N-gram: $$P(w_n \mid w_{n-N+1}^{n-1}) = \frac{C(w_{n-N+1}^{n-1}w_n)}{C(w_{n-N+1}^{n-1})}$$

- 要有一个一致的概率模型，在每个句子中附加一个唯一的开始<s>和结束</s>符号，并将它们视为附加单词。

语言模型的平滑技术（基本概念）

- 由于存在可能的单词序列的组合数，许多罕见（但并非不可能）的组合在训练中从未出现，因此最大似然估计（MLE）错误地将许多参数（也就是稀疏数据）赋值为零。
- 如果在测试过程中出现了新的组合，则给出该组合的概率为零，整个序列的概率为零。
- 在实际处理中，参数被平滑（又称正则化），以重新分配一些概率值给未知的事件。
- 为未知的事件添加概率值需要将其从看到的事件中移除(折现)，以保持和为1的联合分布。

加法平滑技术（了解）

- 对所有事件的频率值加上固定值避免零概率事件，add-δ平滑在每个事件的出现次数加上一个数δ，即

$$P_{add}\left(w_{i-N+1}^{i-1}\right)=\frac{c(w_{i-N+1}w_{i-N+2}\ldots w_i)+\delta}{\sum_{w_i}c(w_{i-N+1}w_{i-N+2}\ldots w_{i-1})+\delta|V|}$$

- 其中$0<\delta\leq 1$，$|V|$是Unigram的词表个数。当δ=1时，称为“加1法”。
- 性能较差。

线性插值平滑技术（了解）

- 利用低阶模型对高阶N-gram模型进行线性插值，平滑公式为：

$$P_{interp}\left(w_i \middle| w_{i-N+1}^{i-1}\right) = \lambda_{w_{i-N+1}^{i-1}} P_{ML}\left(w_i \middle| w_{i-N+1}^{i-1}\right) + \left(1 - \lambda_{w_{i-N+1}^{i-1}}\right) P_{interp}\left(w_i \middle| w_{i-N+2}^{i-1}\right)$$

其中$\lambda_{w_{i-N+1}^{i-1}}$为插值系数

- N元文法语言模型可以递归地定义为由最大似然估计原则得到的N元文法语言模型和（N-1）元文法语言模型的线性插值。

N-gram语言模型的问题：长距离依赖（了解）

- 很多时候，本地上下文并不提供最有用的预测线索，而是由远程依赖提供的
  - 语法的依赖性
    - “The ***man*** next to the large oak tree near the grocery store on the corner **is** tall.”
    - “The ***men*** next to the large oak tree near the grocery store on the corner **are** tall.”
  - 语义依赖性
    - “The ***bird*** next to the large oak tree near the grocery store on the corner **flies** rapidly.”
    - “The ***man*** next to the large oak tree near the grocery store on the corner **talks** rapidly.”
- 需要更复杂的语言模型来处理这种依赖关系

第三章　语音识别

- 3.1 语音识别概述
- 3.2 声学模型
- 3.3 语言模型
  - 3.3.1 语言模型的基本概念
  - 3.3.2 N-gram语言模型
  - 3.3.3 神经语言模型
- 3.4 语音识别解码算法
- 3.5 语音识别技术的展望

神经语言模型

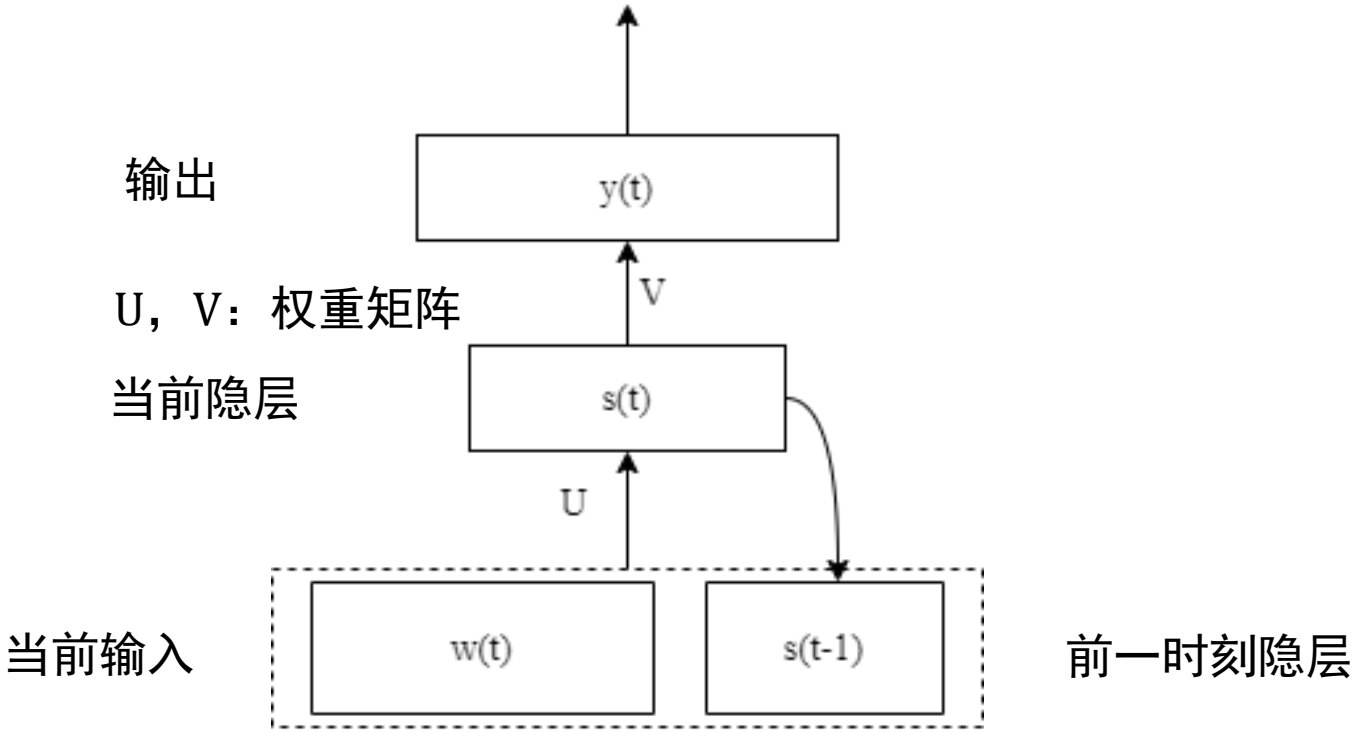
- 神经语言模型（Neural Language Model，NLM）可以用来对文本的长距离依赖建模。
- 不同于基于类的N-gram模型，神经语言模型在能够识别两个相似的词，并且不丧失将每个词编码为彼此不同的能力。
- 神经语言模型共享一个词（及其上下文）和其他类似词。

神经语言模型历史

- 2003年Bengio等人提出了前向神经网络（Feed-forward Neural Network, FNN）语言模型
- 2010年Mikolov等人将循环神经网络（RNN）用于语言模型
- 2013年Graves等人提出基于长短期记忆（Long Short Term Memory, LSTM）循环神网络的语言模型
- 2014年Bahdanau等人[46]首次将注意力机制用于NLP任务
- 2016年Tran等人和Mei等人证明了注意力机制可以提升RNNLM的性能

RNN（循环神经网络）语言模型简介

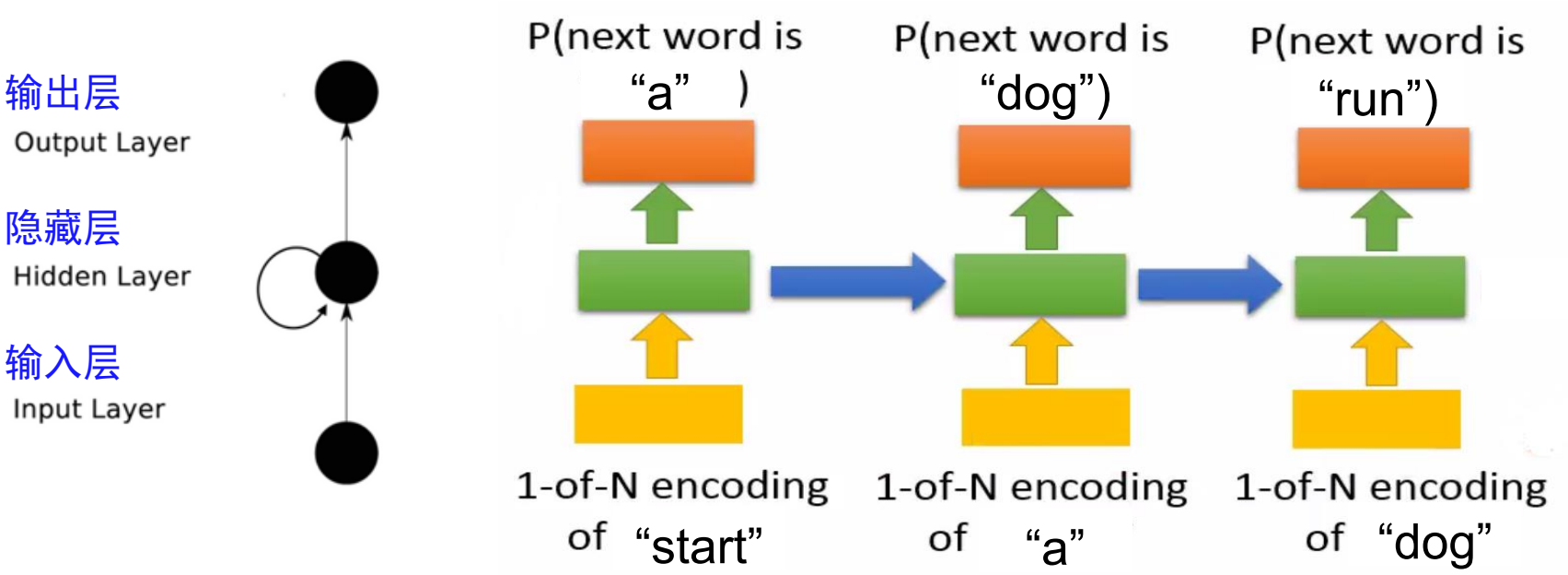
- RNN是一类用于处理序列数据的神经网络，在语言模型中对应单词的前后顺序
- 构建历史长度不受限定，具有更强的记忆能力的网络



RNN的基本结构

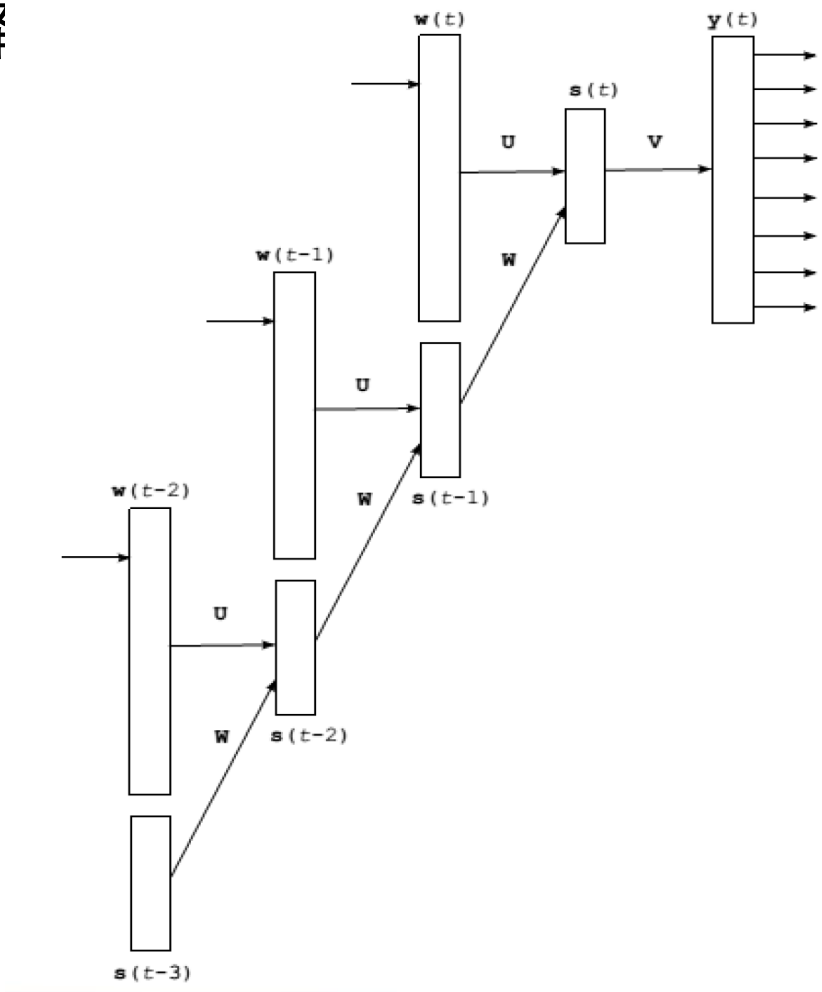
RNN（循环神经网络）语言模型简介（续）

- 例子



RNN（循环神经网络）语言模型简介（续）

- 长上下文建模解释

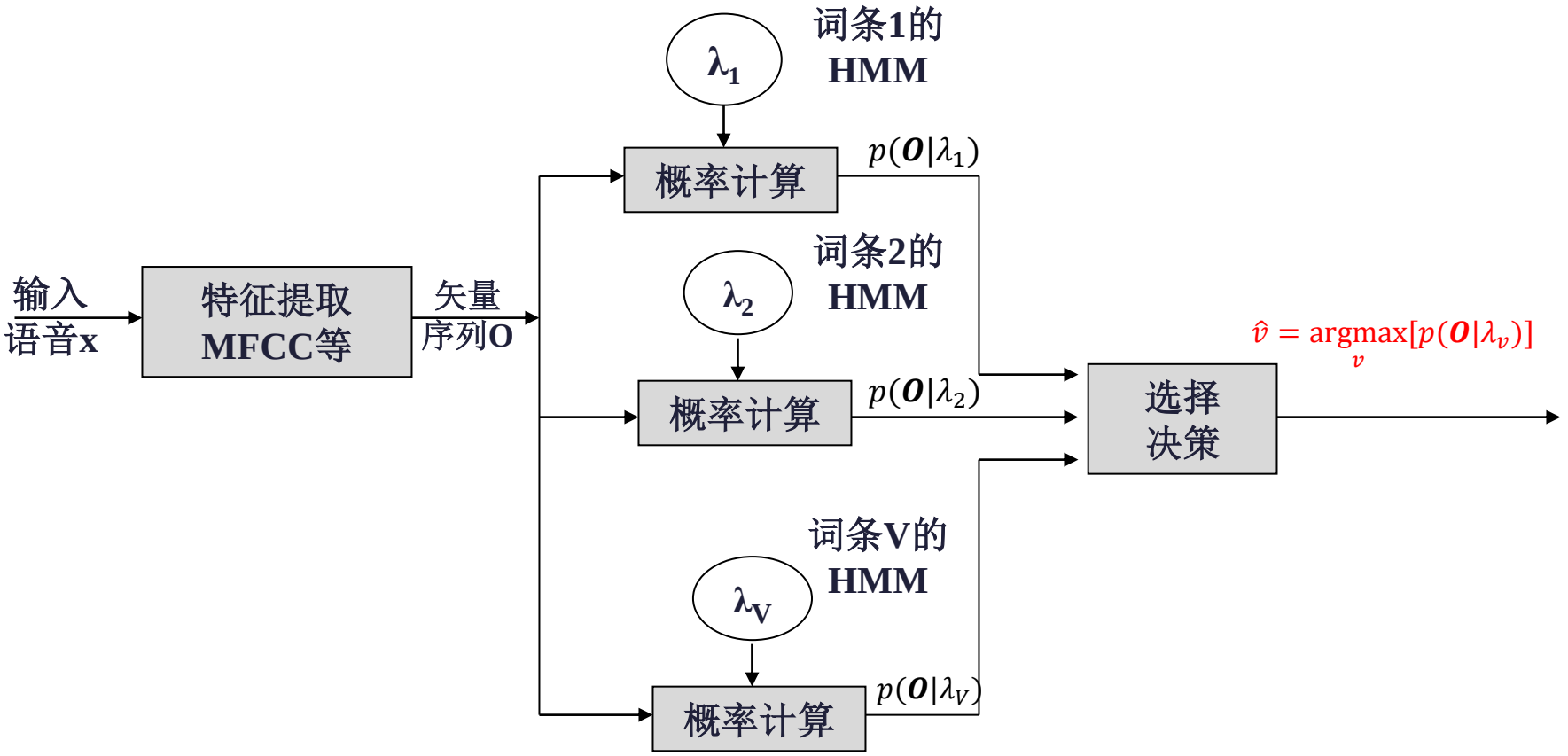


第三章　语音识别

- 3.1 语音识别概述
- 3.2 声学模型
- 3.3 语言模型
- 3.4 语音识别解码算法（了解）
- 3.5 语音识别技术的展望

孤立词识别及解码技术

- 孤立词识别原理图



连续语音识别的解码算法

$$P(W \mid Y) = \arg\max_{\{w_1^N\}\{t_1^N\}} \{ \sum_{n-1}^{N} \log(P_{acoust}(y_{t_{n-1}+1}^{t_n} \mid w_n)) + \lambda \cdot \sum_{n-1}^{N} \log(P_{lang}(w_n \mid w_{n-1}) + \delta) \}$$

$P_{acoust}(y_{t_{n-1}+1}^{t_n} \mid w_n)$ ： 声学模型的似然

$P_{lang}(w_n \mid w_{n-1})$ ： 语言模型的似然

λ ： 权重　　δ ： 插入的惩罚

连续语音识别的解码算法

- 对大词汇量连续语音识别，最终目的是从各种可能的子词序列形成的一个网络中，找出一个或多个最优的子词序列。这在本质上属于搜索算法或解码算法的范畴。
- 在搜索过程中，路径数随着搜索的进行急剧增加。全搜索几乎是不可能的。因此常采用基于一定裁剪路径的算法。
- 解码算法
  - 基于Viterbi的解码算法
  - 基于加权有限状态自动机（WFST）

Viterbi beam 搜索解码算法

Viterbi beam搜索算法是一个广度优先的帧同步算法，在每一时刻有效地剪裁低得分路径。

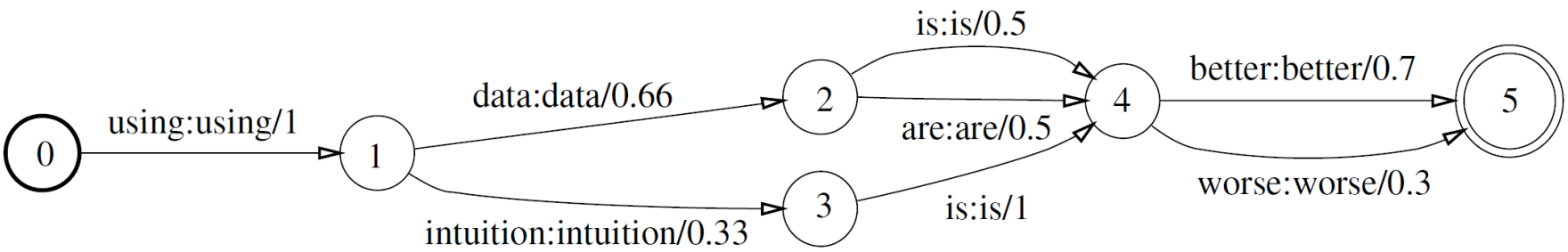
- 初始化
  初始化活动路径（最高层）
- 递推
  For 帧t=1 到T
    For 每一层次（指各个层次的语言和声学模型）
      For HMM的每个活动状态
        把每个活动路径向后扩展一帧至所有可以到达的状态
        执行Viterbi计算
        裁剪路径
      End {活动状态}
    End {每一层次}
  End {观察矢量序列}
- 终止：选择最可能的路径

基于Viterbi的解码算法的问题：

- 由于LVCSR实际任务中N-gram，词典和三音素建模导致无法只简单的使用Viterbi算法，需要进行一些工程优化
  - 剪枝(Pruning)
  - 多阶段解码(Multi-stage decoding)
  - 语言模型超前使用(Language model Look-ahead)

加权有限状态转换机(WFST)

- WFST由状态节点和边组成，且边上有对应的输入、输出符号及权重，形式为 x ： y/w，表示该边的输入符号为x、输出符号为y、权重为w，权重可以定义为概率（越大越好）、惩罚（越小越好）等，从起始到结束状态上的所有权重通常累加起来，记为该条路径的分数，一条完整的路径必须从起始状态到结束状态。



语言模型WFST示例

连续语音识别中的WFST

- 连续语音识别利用几个知识来源（词典、语法、音素、声学模型）寻找最有可能的单词序列。

$$\mathrm{HCLG} = \mathrm{H} \circ C \circ \mathrm{L} \circ G$$

G：语法或语言模型接收器的概率
L：词典（音素或或字或词）
C：上下文相关转换机（将上下文相关的音素转化成上下文无关的音素）
H：HMM转换机

WFST解码

$$\mathrm{W}' = argmax_W P(X|W)P(W)$$

- 求解以上问题可以转换成：
  - 四种WFST组成一个大的解码图$\mathrm{H} \circ C \circ \mathrm{L} \circ G$，然后在这个解码图上进行搜索，将HMM状态转换成最终的句子。
  - 将特征向量X用HCLG对齐来解码，

$$\mathrm{W}' = argmax_W \mathrm{X} \circ （\mathrm{H} \circ C \circ \mathrm{L} \circ G）$$

第三章　语音识别

- 3.1 语音识别概述
- 3.2 声学模型
- 3.3 语言模型
- 3.4 语音识别解码算法
- 3.5 语音识别技术的展望
  - 3.5.1 端到端语音识别（了解）
  - 3.5.2 主要挑战（掌握、了解）

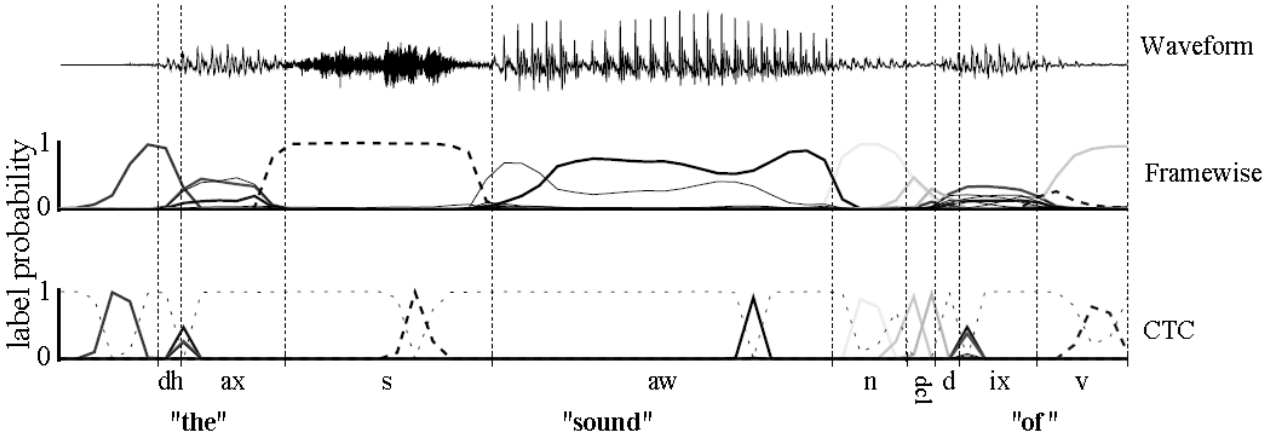
基于CTC的语音识别算法

- 什么是CTC？
  - Connectionist temporal classification，基于神经网络的时序信号分类。

- 语音识别为什么需要CTC？
  - 传统的语音识别的声学模型训练，对于每一帧的数据，需要知道对应的标签(三音素)。
  - 必须预先训练一个GMM-HMM声学模型。
  - 采用GMM-HMM系统进行自动对齐标注。
  - 对齐的准确性严重依赖GMM-HMM声学模型的可靠性。
  - 不论是GMM-HMM声学模型训练，还是语音对齐过程，都需要进行反复多次的迭代，非常耗时。
  - 分阶段训练不合理。

基于CTC的语音识别算法

- 引入’blank’ 符号



- 对于一段语音，CTC最后的输出仅是spike（尖峰）的序列

“--a-bb”→ “ab”

- 每个CTC输出序列的概率实际上是多个填充了blank符号后输出序列的概率的总和

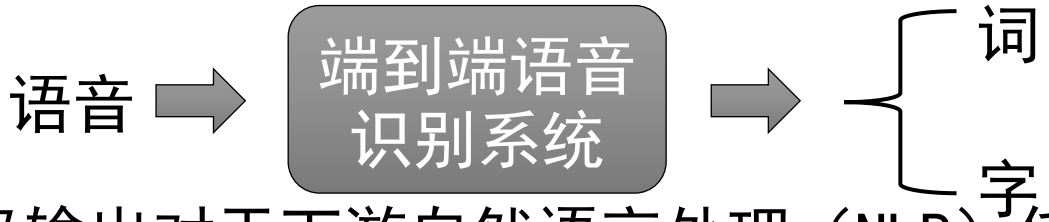
基于Attention的语音识别算法（基本概念）

- 另一个目前常用的序列到序列的模型是编码器-解码器模型（Encoder-Decoder Model）
- 该模型由编码器和解码器两部分组成，其中编码器将输入序列映射到一个指定长度的向量，解码器将指定长度的向量解码为真实的标签。
- 2014年[1]提出将Attention机制加入到目前的编码器-解码器模型（Encoder-Decoder）中，并成功应用到了机器翻译中，同时取得了目前最好的效果。

[1] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[J]. arXiv preprint arXiv:1409.0473, 2014.

基于Attention的语音识别算法

- 端到端语音识别的优点
  - 直接将声学特征映射到更大的单元序列中，而不是音素或初始序列-最终序列



  - 字级输出对于下游自然语言处理（NLP）任务来说很容易
- Encoder-decoder框架存在如下问题
  - 长序列对编码器网络造成巨大的压力
  - 解码器网络需要从固定长度的编码向量中寻找与当前时刻相关的序列内容，如果编码向量中压缩太多信息，会导致解码器网络很难准确找到与当前输出对应的序列信息

基于Attention的语音识别算法（基本概念）

- Attention模型的优点
  - 与传统的混合模型相比，Attention的模型实现了端到端训练，能够将整个系统训练到最优。
  - 与CTC相比，Attention不需要每帧独立的假设。
- Attention模型的缺点
  - 在噪声环境下很难训练到很好的效果，通常要比CTC的方法差
  - 目前的解决方案：CTC和attention联合训练，采用CTC损失函数针对下层Encoder做强制对齐。

第三章　语音识别

- 3.1 语音识别概述
- 3.2 声学模型
- 3.3 语言模型
- 3.4 语音识别解码算法
- 3.5 语音识别技术的展望
  - 3.5.1 端到端语音识别
  - 3.5.2 主要挑战

复杂声学场景下的语音识别（掌握）

- 当语音识别系统处于复杂声学场景下，由于训练声学模型的声学场景与真实复杂声学场景差异过大，导致语音识别系统性能大大降低。

- 而现实生活中，声学场景千变万化，训练集不能完全覆盖所有声学场景

- 例如
  - 训练时噪声、混响等环境可穷举、可模拟，现实中无法穷举，导致训练集和测试集不匹配

复杂声学场景下的语音识别（掌握）

- 主要挑战
  - 真实世界传感器需要获取声音，但是周围声音是各种各样的，需要获取有效的人声
  - 获取人声也不一定是目标任务的人声，可能是用户和别人交谈的声音
  - 在远场场景下获取的声音面临着回声干扰、室内混响、多信号源干扰以及非平稳噪声的干扰等等
- 现状
  - 研究新的声学信号处理技术或新的声学信号处理与语音识别的联合优化方案

低数据资源语音识别（了解）

- 低数据资源是指音频或者有转录文本的音频较少，导致语音识别性能下降
- 全世界共使用5651种语言，其中，汉、英、印度、俄、西班牙、德、日、法、印度尼西亚、葡萄牙、孟加拉、意大利和阿拉伯语是使用人数较多的语言
- 其他一些少数人使用的语言在语音资源上就显得匮乏（可能只能获得几十小时的音频）

低数据资源语音识别（了解）

- 主要挑战
  - 端到端语音识别系统需要大量训练数据，有些特定领域的语音很难获取，包括但不限于以下人群：
    - 非母语学习者
    - 构音障碍患者（由于疾病导致发音的错误，产生的语音可能扭曲、错误）
    - 少数民族人群
- 现状
  - 研究基于迁移学习、模型自适应等算法

多语言和跨语言语音识别（了解）

- 目前大多数语音识别系统都是针对一种语言的
- 对于每种语言都训练一个语音识别系统，所需要的成本太高，且需要大量的数据
- 多语言语音识别和跨语言语音识别的区别在于
  - 多语言语音识别只能识别训练集中出现过的语言
  - 跨语言语音识别能识别出从未见过的语言。
- 例子
  - 多语言语音识别：一个用英语、德语训练出来的模型，可以同时对英语、德语进行语音识别
  - 跨语言语音识别：一个用英语、德语训练出来的模型，可以同时对英语、德语、法语进行语音识别

多语言和跨语言语音识别（了解）

- 主要挑战
  - 如何选择建模单元。像字符、子词等单元很难扩展到词汇量大的语言
  - 词汇量大导致标签稀疏问题
  - 构建发音字典需要每个语言专家的知识，要耗费很大的人力
  - 有些语言很难获取
- 现状
  - 需要新技术、新的解决方案

语种混杂语音识别（了解）

- 在我们日常交流中，经常会有中文语境下英文单词夹杂的现象，在学术上称为语种混杂（Code-switch）

- 例子

  一般语音识别对于句子“这里有Wi-Fi吗”，可能会识别为“这里有外海吗”

语种混杂语音识别（了解）

- 主要挑战
  - 由于嵌入语（英文）受主体语（中文）影响形成的非母语口音现象严重
  - 不同语言音素构成之间的差异给混合声学建模带来巨大困难
  - 带标注的混合语音训练数据极其稀缺

- 现状
  - 最近，端到端的语音识别算法与技术逐步被应用到语种混杂语音识别任务上

多口音语音识别（了解）

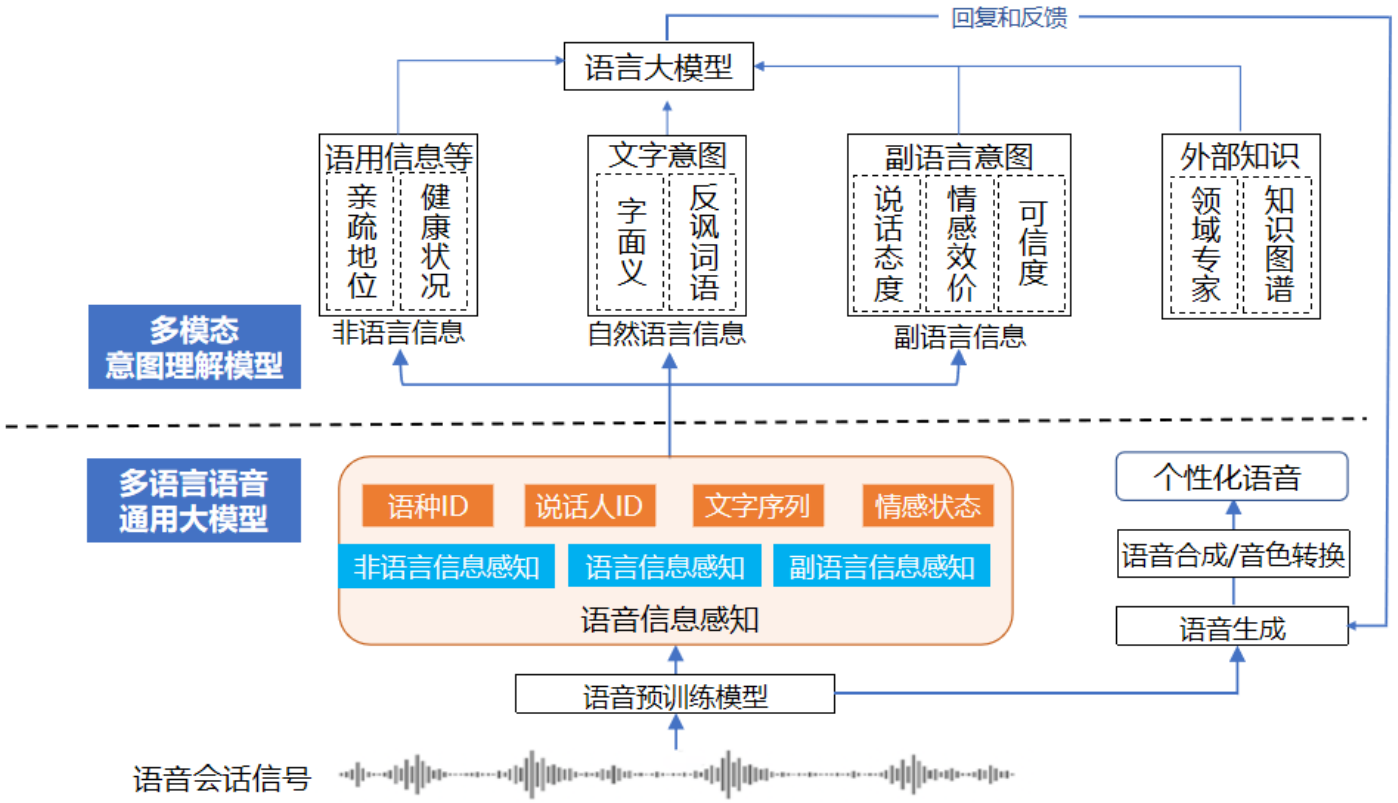
- 在日常生活中，一个地区的人在说另一个地区的语言时，容易保持自己习惯的发音方式
- 以汉语为例，由于不少普通话使用者把普通话作为第二语言来掌握
- 发音不可避免地受到其方言母语发音的强烈影响，出现发音不准确、发音错误等现象，导致语音识别性能下降

多口音语音识别（了解）

- 主要挑战
  - 受母语影响不同方言背景的说话人的发音具有差异性
  - 可用于训练语音识别系统的多口音的数据极其稀缺

- 现状
  - 对于第一个问题，虽然不同方言口音之间存在着差异，但也存在着相似性，因此，研究如何改进声学模型以适应更多变化的口音是一种研究趋势
  - 第二个问题也是现在语音识别的难点之一

声学模型与语言大模型的联合建模（了解）



“海河·谛听”言语交互意图理解大模型

- Encoder：如何兼顾声学信息和语义信息的提取；如何充分利用海量的无标注数据；
- Decoder：如何更好地融合声学特征和文本特征；如何从声学特征中分解出语言信息、非语言信息和副语言信息；如何将额外的海量文本知识引入语音处理任务等。

上机实践课

内容：语音识别系统实现及验证
时间：9月28日、10月7日、10月9日
机房：47教2号机房