语种识别

Friday 11th January, 2019

1 什么是语种识别

生活在校园里每天都能听到各种不同的语言：普通话、英语、东北话、天津话、粤语、闽语，还有偶尔听到的Indialish。只要我们脑子里有这种语言的记忆，不论我们懂不懂是什么意思，我们都可以区分我们听到的每种语言，并立刻判断出来这是什么话。我们似乎与生俱来就有这样的超能力。

而对计算机来说，它并不具有这种超能力。计算机对语种的识别能力是在最近六十年才发展起来的。对计算机来说，可以将语种识别看作是模式识别问题。语种识别最早是1974年由Leonard 和Doddington提出[1]的。但是直到上世纪90年代美国国家标准技术研究院开始组织的语种识别评测比赛，语种识别技术才得以快速发展。目前语种识别方法主要分为基于音素识别器的语种识别、基于底层声学特征的语种识别和基于深度学习的语种识别。

一个典型的语种识别系统主要分为三步，分别是特征提取、模型建立和模型分类，如Figure 1 所示。

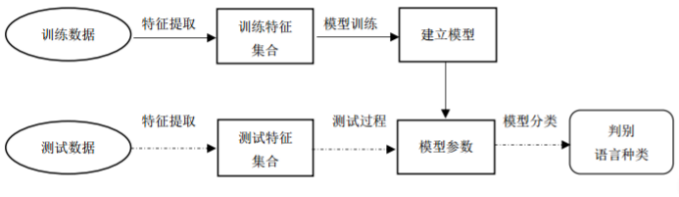

Figure 1: 语种识别基本流程图

研究结果表明，人类自身在进行语种识别时常用的区分性特征按照从低层到高层可以划分为底层声学特征（MFCC、SDC、PLP等）、韵律特征（时长、基频、重读等）、音素（音素级别的N-gram）、词法和语法[2]。底层声学特征包括频谱、倒谱、谱包络以及共振峰，被认为是声学信号所固有的物理特征。其

它特征都是建立在底层声学特征基础之上的更高层次的特征，来自于低层信息的分析和提取。越往高层，特征区分语种的能力越强，然而特征提取的代价也越高。越往底层，特征中所含的冗余信息越多，特征区分语种的能力越弱，但是特征提取的代价越低。语言学上的研究表明，语言中可用于语种分类的特性在每一层信息中都有着不同的体现。目前大部分语种识别研究集中在底层的声学特征和音素特征上。

语种识别虽然已经发展了近六十年，在性能上也取得了很大的进步，但仍然面临着严峻的挑战，主要包括外界噪声、短时语音需求和易混方言或者口音识别任务。语种识别任务正向着复杂场景、实时性以及精细语言种类识别转变。基于现有的声学特征建模方法与基于音素识别的建模方法都难以取得重大进展，主要原因在于底层声学特征对于噪声鲁棒性的影响极易受到说话人、信道、环境噪声以及特定说话内容的干扰，语音识别器又受限于场景条件下语音转录的语音数据是否可使用问题，从而影响特征对语种的区分效果。这迫使研究者们开拓新的思路与方法。最近几年，深度学习(Deep Learning)理论在语音识别领域取得了令人振奋的成果，迅速成为了当下学术界和产业界的研究热点，为处在瓶颈期的语音识别领域提供了一个强有力的工具，所以产生了基于深度学习的语种识别的热潮。

2 基于音素识别器的语种识别方法

基于音素识别器的语种识别方法认为不同的语言之间相同的音素搭配体现的统计特性是有差别的，因此可以用来进行语种识别。具体实现而言，首先使用音素识别器从语音数据中提取语音信息，该音素识别器给出一系列音素标签，在此之后，使用N-gram语言模型(Language Model，LM)来估计在每种目标语言中出现特定音素序列的概率。估计每种语言中出现的所有可能的音素序列的概率的N-gram可以给出语言模型LM，其捕获关于该语音的语言信息。该方法在语种识别领域被称为音素识别器结合语言模型(Phone Recognizer followed by Language Model，PRLM) [3] [4]。

为了充分利用不同阶数的描述能力，克服语种中不同阶N-gram因语料过少而造成的稀疏性问题，研究者在语言模型的基础上提出了二叉决策树(Binary Tree，BT)模型，称为音素识别器结合二叉决策树(Phone Recognizer followed by Binary Tree，PRBT)方法。由于每个语种的音素单元不一致，因此语种的N-Gram差异在不同的语种音素识别器上的反映有所不同。有研究者提出使用多个音素识别器作为前端来提取同一语音在不同语种音素集合下的音素序列，然后在判决得分上对所有子系统进行融合。该方法称之为并行音素识别

器(Parallel Phone Rcognizer followed by Language Model，PPRLM) 方法，相应的就产生了并行音素识别器结合语言模型(PPRLM) 和并行音素识别器结合二叉树模型(PPRBT)。

随着支持向量机模型(SVM) [5]的发展，研究者们又提出利用N-gram基元构建词袋矢量bag-of-Ngram，利用SVM来构建模型的PPRSVM方法。它具有建模能力更强，对小样本情况鲁棒性更强的优点，成为了目前基于音素识别器方法中的主流系统。在SVM之后，大量的研究工作都集中在解决N-gram的稀疏性以及挑选出最具有区分性的N-gram的单元等问题。

3 基于底层声学特征的语种识别方法

基于底层声学特征的语种识别方法是利用底层声学特征所能够描述的声学单元的统计特性差异来对语种进行分类。由于该方法所需要的特征直接通过底层的谱参数得到，不需要音素识别器作为支撑，因此一直以来都是语种识别研究的热点。该种方法在近年来取到了以下三大突破：

第一大突破就是2002年提出的SDC特征[6]结合混合高斯模型-通用背景模型(Gaussian Mixture Model-Universal Background Model，GMM-UBM)的方法。基于此种方法，大量的改进方法不断被提出。研究者们提出抑制卷积信道噪声影响的倒谱域减均值(Cepstral Mean Subtraction，CMS)、去掉说话人影响的声道长度规整(Vocal Tract Length Normalization，VTLN)、特征高斯化以及RASTA滤波等方法，用以克服底层声学特征易受说话人、信道、噪声以及内容信息差异影响的问题。

第二大突破就是区分性建模方法的引入。传统的GMM-UBM模型是一个典型的生成性模型，当它处理不同类别之间的易混部分时效果较差。因此在GMM的基础上，研究者们分别提出了不同的解决办法：利用区分性训练准则最大互信息准则(Maximum Mutual Information，MMI)训练语种GMM模型的GMM-MMI方法；利用区分性的SVM模型来对对每段语音的GMM均值超矢量进行建模的GSV-SVM方法；利用GSV-SVM方法反推语种GMM模型的model pushing方法。

第三大突破就是因子分析方法(Factor Analysis，FA)。该方法受到说话人识别当中联合因子分析方法(Jiont Factor Analysis，JFA)的启发，在底层声学空间中对信道噪声进行子空间建模，然后通过特征域的去噪或者模型域的补偿去除噪声的影响。由于构建的子空间中仍然包含着有效分类信息，研究者们上提出了基于全差异空间建模的方法(Total Variability，TV)。该方法围绕样本的GMM超向量与均值超向量之间的差异，将每个样本视为独立的个体，训

练得到每个样本之间的全差异空间，然后得到样本之间差异的低维表示称之为i-vector。之后通过线性区分性分析或者类内协方差规整等技术对i-vector进行类内类间差异补偿和降维，再采用SVM或者快速余弦距离来进行建模。目前，TV[7]方法(或者称为i-vector方法)因其低维的语音段表示以及良好的性能成为了语种识别领域的主流系统。

3.1 SDC特征

Delta和Delta-Delta倒谱有效地包括时间信息，但是它们在模拟语音的更高级时间的能力方面受到限制，因为它们仅在当前时间点模拟倒谱的斜率。使用标准的N=2值计算方法，delta倒谱将是基于5帧（50 ms）的值估计当前时间的斜率。因此，充其量，它们仅能够在50ms的时间窗内结合语音的时间信息。

SDC是在较长时间窗口中将时间信息包括在语音信号中的更好的替代方案。SDC主要是在底层谱参数特征MFCC或者PLP的基础上通过移位差分扩展而来。一个典型的SDC特征提取流程如Figure 2所示。

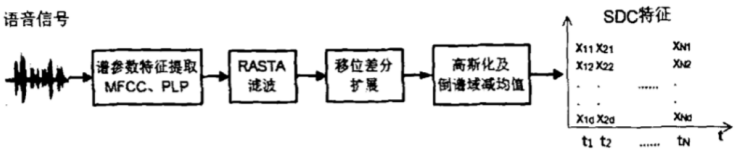

Figure 2: 典型的SDC特征提取流程框图

首先对语音加窗提取其底层声学谱参数特征，在语种识别任务中一般提取MFCC特征或者PLP特征。在提取完谱参数特征后，采取RASTA滤波(Relative Spectral filtering)来抑制参数表示中非语音频谱部分的影响。对于提取出来的第t帧静态谱特征N维静态谱参数特征c(t)，其对应的第k个一阶差分矢量计算表达式为

$$\Delta c(t,k) = c(t+(k-1)P+d) - c(t+(k-1)P-d) \tag{3.1}$$

此时，得到的SDC特征就是将静态特征和k个移位差分矢量$\Delta c(t,k)$拼接起来，形成最终的SDC 特征x_t。

$$x = \begin{bmatrix} c(t) \\ \Delta c(t,0) \\ \Delta c(t,1) \\ \cdots \\ \Delta c(t,k-1) \end{bmatrix} \tag{3.2}$$

一个典型的移位差分计算如Figure 3所示，它的计算主要由4个参数N-d-P-k描述。其中N表示提取出来的静态参数特征维数，d表示计算一阶差分的帧与参考帧的时移距离，P表示参考帧的跳帧长度，i表示参考帧的个数，由于其在每个参考帧下都会得到一个差分矢量，因此也表示为一阶差分矢量单元数目。

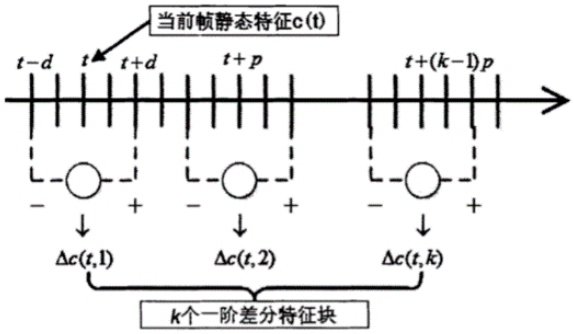

Figure 3: 移位差分计算示意图

在得到SDC特征后，一般还采用倒谱域件均值(CMS)来去除信道的卷积噪声以及特征高斯化技术对语音的参数进行规整，这些技术就构成了目前SDC特征提取的标准流程。这样就得到了一段语音s的D维帧级(Frame-level)特征表示X=X_1，...，$X_T n$，其中T_s 表示语音段s的总帧数。

3.2 GMM-UBM语种识别方法

UBM其实就是一个大型的GMM模型，用来训练表示与说话人无关的特征分布。它的训练数据是某一信道下的所有人的语音数据，而不是想target模型只是反映某一个人的特征分布。说白了，只是一个大的GMM，那么训练UBM也就是训练GMM，所用算法采用的是EM算法。

GMM中，从说话人语音抽出来的D维特征矢量对应的似然率可用K个高斯分量表示：

$$p(x|\lambda) = \sum_{i=1}^{M} w_i p_i(x) \tag{3.3}$$

其中是第K个高斯分量的权重，$\sum_{i=1}^{M} w_i = 1$

$$p_i(x) = \frac{1}{(2\pi)^{D/2}|\sum_i|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu_i)'(\sum_i)^{-1}(x-\mu_i)\right\} \tag{3.4}$$

所以，整个高斯模型可以用模型参数$\lambda = \{w_i, \mu_i, \sum_i\}$，i=1,2,...,m来表示。

3.3 GMM-MMI语种识别方法

由语音识别中区分性训练的成功应用启发，研究者们开始将区分性训练准则引入到GMM模型的训练领域，其中基于最大互信息准则(MMI)的训练方法最为成功。最大互信息准则实质上是最大化训练数据真实类别的后验概率。假设训练集合语音段共有N句S=s_1 , . . . ，s_n，对应的每句语音段提取出来的特征集合表示为X=X_1, . . . ，X_n，其中Xn=$x_n, 1$, . . . ，x_n, Tn)，那么MMI 准则的目标函数可以表示为

$$L_{MMI}(\Lambda_l|\chi) = \frac{1}{N}\sum_{n=1}^{N}\frac{p(\chi_n|\Lambda_{ln})|p(l_n))}{\sum_l p(\chi_n|\Lambda_l)p(l)} = \frac{1}{N}\sum_{n=1}^{N}p(\Lambda_{ln}|\chi_n) \tag{3.5}$$

其中l_n表示语音段s_n真实的语种标记，$p(\lambda_{ln}|X_n)$表示语音段s_n 的属于真实类别ln的后验概率，p(l)表示语种l的先验概率，一般而言认为语种的是先验是等概率的，因此一般可以忽略掉。

SDC和GMM模型的出现，促进了基于底层声学特征语种识别方法。由于它不像PR方法那样对识别器有很强的依赖，建模方法更加容易实现，大量的研究者们开始致力于基于声学特征的语种建模方法研究，这期间区分性建模方法和因子分析方法的引入是两个最具影响力的进展。

3.4 GSV-SVM语种识别方法

高斯均值超矢量(GMM Super Vector, GSV)来源于GMM-UBM。与SVM组成GSV-SVM模型，广泛用于语种及说话人识别。GSV以GMM模型的均值（或者方差）超矢量作为输入特征序列，避免了直接使用带噪语音信号的特征参数，在实际应用中获得了更好的实验效果。

利用前文GMM-UBM模型，通过MAP自适应获得代表语音段的GSV特征。GMM均值超矢量的计算过程如Figure 4：

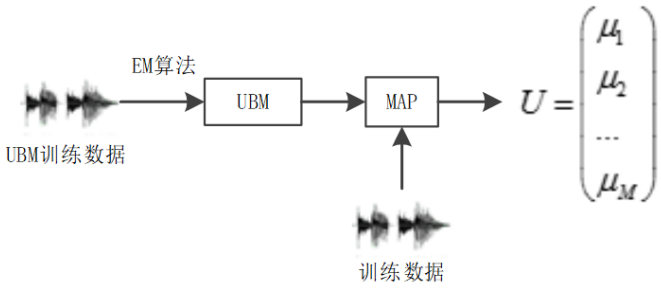

Figure 4: GSV计算过程

MAP自适应的过程，是根据目标语种的训练特征向量与UBM的匹配程度，将UBM的各个高斯混元向目标语种模型“拉近”的过程。对于目标语种的训练数据O=(o_1,o_2,...,o_T)，首先计算o_t 与UBM 中每个高斯模型的匹配似然度，见式(3.6)：

$$P(m|O_t,\lambda_{UBM})=\frac{w_m P_m(O_t|\mu_m,\sum_m)}{\sum_{i=1}^{M} w_i P_i(O_t|\mu_i,\sum_i)} \tag{3.6}$$

然后利用P(m—o_t ， λ_{UBM})和o_t分别计算对混合权重、均值矢量和均方值的充分估计，见式(3.7)：

$$\begin{cases} n_m=\sum_{t=1}^{T} P(m|O_t,\lambda_{UBM}) \\ E_m(O)=\frac{O_t}{n_m}\sum_{t=1}^{T} P(m|O_t,\lambda_{UBM}) \\ E_m(O^2)=\frac{O_t^2}{n_m}\sum_{t=1}^{T} P(m|O_t,\lambda_{UBM}) \end{cases} \tag{3.7}$$

然后利用这些充分统计和修正因子对第个高斯的参数进行修正，具体过程见式(3.8)：

$$\begin{cases} \bar{W_m}=[a_m^w n_m/T+(1-a_m^w n_m)w_m]\gamma \\ \bar{\mu_m}=a_m^\mu E_m(o)+(1-a_m^\mu)\mu_m \\ \bar{\sigma_m^2}=a_m^v E_m(o^2)+(1-a_m^v))(\sigma_m^2+\mu_m^2)-\bar{\mu_m^2} \end{cases} \tag{3.8}$$

其中，γ为权重的规整因子，用来保证$barW_m$的和为1,a_m^w、a_m^u、a_m^v 为第m个高斯的权重、均值和方差的修正因子，见式(3.9)：

$$a_m^\rho=\frac{n_m}{n_m+r^\rho},\rho\in w,\mu,v \tag{3.9}$$

其中，r^ρ为常数，用来控制修正因子的变化尺度，一般取为16。由于自适应语音不够长，无法准确描述每个高斯的权重和方差，MAP过程中，一般只更新均值，权重和方差保持和原来的UBM模型一致。

4.2.5 I-vector特征

语种语音的帧级局部特征MFCC作为i-vector特征提取步骤的前端输入。为了抑制信道噪声，对MFCC差分特征进行谱均值方差归一化处理(Cepstral Mean Variance Normalization, CMVN)。具体提取步骤如Figure 5 所示：

首先利用部分训练数据通过期望最大化(Expectation Maximum, EM)得到UBM。然后利用最大后验概率(Maximum A Posterior, MAP)自适应得到GMM，根据JFA理论，重新定义GMM均值超矢量，见式(3.10)：

$$M=m+Tw \tag{3.10}$$

其中，M表示GMM均值超矢量，m表示一个与特定目标方言和信道都无关的超矢量，通常由UBM均值超矢量替代。全差异载荷矩阵(Total Variability

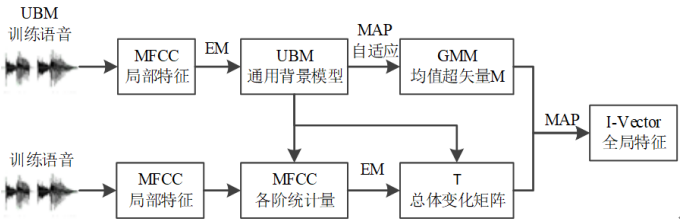

Figure 5: i-vector特征提取流程图

Matrix)的估计是关键性环节，首先计算语音MFCC特征相对于UBM 均值超矢量m 的零阶N_c、一阶F_c^1及二阶F_c^2统计量，见式(3.11)、(3.12)、(3.13)：

$$N_c = \sum_{t=1}^{L} P(c/y_t, \theta_{UBM}) \tag{3.11}$$

$$F_c^1 = \sum_{t=1}^{L} P(c/y_t, \theta_{UBM})(y_t - m_c) \tag{3.12}$$

$$F_c^2 = \sum_{t=1}^{L} P(c/y_t, \theta_{UBM})(y_t - m_c)(y_t - m_c)^T \tag{3.13}$$

其中，c=1,2,..., C，m_c与$P(c/y_t, \theta_{UBM})$分别为UBM第c个高斯子分布的均值及后验概率。其次，利用各阶统计量，通过EM算法随机初始化全差异矩阵，在最大似然准则(Maximum Likelihood, ML)下估计w(即I-vector)的一阶和二阶统计量，见式(3.14)、(3.15)：

$$E_s^1(w) = L_s^{-1} T^T \sum^{-1} F_s^1 \tag{3.14}$$

$$E_s^2(ww^T) = E_s^1(w) E_s^1(w^T) + L_s^{-1} \tag{3.15}$$

其中，L_s是临时变量，具体表示见式(3.16)：

$$L_s = I + T^T \sum^{-1} N_s T \tag{3.16}$$

N_s是由N_c作为主对角元拼接得到的矩阵，F_s^1是由F_c^1 直接拼接得到的矢量，I 是单位矩阵，$\sum$是UBM的协方差矩阵。T 和$\sum$的更新见式(3.17)、(3.18)：

$$\sum_s N_s T E_s^2(ww^T) = \sum_s F_s^1 E_s^1(w) \tag{3.17}$$

$$\sum = N^{-1}\sum_{s} F_s^2 - N^{-1}diag\left\{\sum_s F_s^1 E_s^1 (w^T T^T)\right\} \tag{3.18}$$

其中，$N = \sum N_s$，F_s^2由F_c^2进行矩阵拼接得到。上述步骤反复迭代6 8 次后，近似认为T 和$\sum$收敛。假定GMM 的高斯子分布数为C，MFCC特征的维数为D，i-vector的维数为K，那么超矢量M和m的维数是C x D，全差异空间T就是CD X K的矩阵。i-vector特征矩阵计算见式(3.19)：

$$w = (I + T^T \sum^{-1} N_c T)^{-1} T^T \sum^{-1} F_c^1 \tag{3.19}$$

通过TV法提取的i-vector特征整合了帧级局部特征MFCC，以语音段为单位表征信息且与语音段长度无关。

4 基于深度学习的语种识别方法

2006年，Hinton在《科学》上发表的一篇文章提出深度神经网络模型的训练方法，掀起了深度学习在学术界及工业界的热潮。深度学习方法在语音识别、图像处理、机器翻译等领域应用广泛，各种深度网络改进算法也相继提出，为相关学术领域提供新的思路和模型，成为研究学者关注的热点。

特别是在语音识别领域，DNN模型给处在瓶颈阶段的传统的GMM-HMM模型带来了巨大的革新，使得语音识别的准确率又上了一个新的台阶，目前国内外知名互联网企业(谷歌、讯飞以及百度等)的语音识别算法都采用的是DNN方法。

本节主要介绍基于深度学习的语种识别的常见方法及其所用模型的优缺点。

4.1 DNN模型

研究者们早在2014年就发现当提供大量训练数据时，使用DNN来解决自动语音识别（LID）任务中Cavg的百分比可高达70[24]，具体实现网络结构如Figure 6 所示,将语音信号首先进行特征预处理得到声学特征，再将声学特性传送至DNN模型中，利用softmax得出其概率向量判别该段语音所属类别。该模型适用于较大的数据量，但其鲁棒性大大提高。

除了将DNN用于语种的分类模型域，Maryam Najafian，Sameer Khurana等人[13]还利用DNN 进行特征提取。基于两个连续的深度神经网络（DNN）ASR模型可提取i-vector特征，流程图如Figure 7 所示，第一个DNN的输入包括从梅尔滤波器组获得的23个临界频带能量，第二个DNN 的输入特征是从第一个DNN 输出的SSD层。

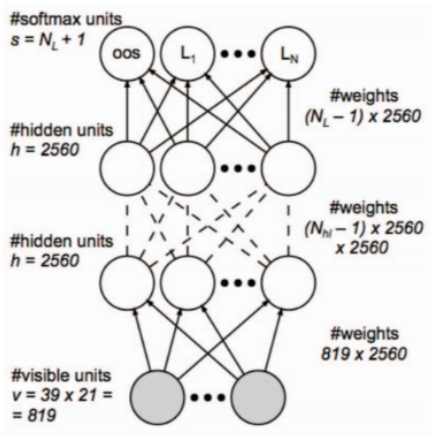

Fig. 1. DNN network topology

Figure 6: DNN network topology , 图来自[24]

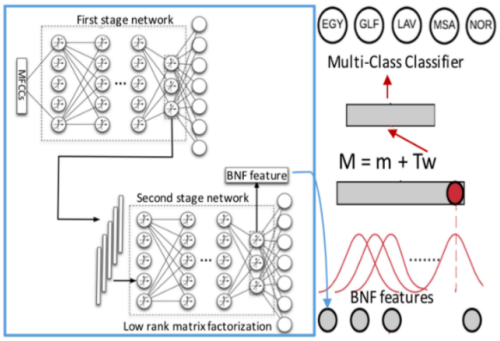

Figure 3: *I-vector based DID system*

Figure 7: DID中提取i-vector模型, 图来自[12]

4.2 RNN模型

4.2.1 RNN

在[14]中研究了端到端的RNN及其变种LSTM,GRU的语种识别。由于RNN存在收敛问题，导致训练网络经常面临消失梯度和爆炸梯度问题的问题。传统的基于RNN神经结构的端到端LID系统如图4-3所示，该系统包含三个主要模块。首先，特征预处理器从原始信号中提取声学特征，如Filter banks。其

次，将提取出的以帧为编码的语音特征输入至RNN网络结构中。再次，使用softmax激活函数概率解码器将RNN的隐藏状态投影到目标语言的可解释概率向量中。最后，通过对比概率的大小判决最可能的种类。此外，还有一种将后验概率转换为对数似然比（LLR）的方法，这适应了更灵活的决策过程。

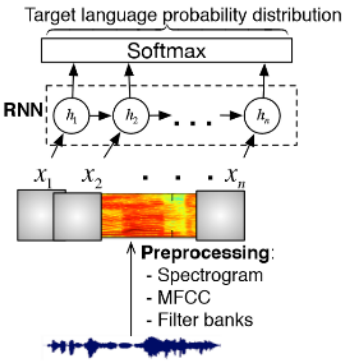

Figure 1: *Design for an end-to-end LID system using RNN.*

Figure 8: RNN结构的端到端LID模型, 图来自[14]

4.2.2 LSTM

同样，在[15]中，研究者们也利用LSTM进行语种分类，LSTM网络结构如Figure 9所示，其解决了RNN中存在的梯度爆炸问题。

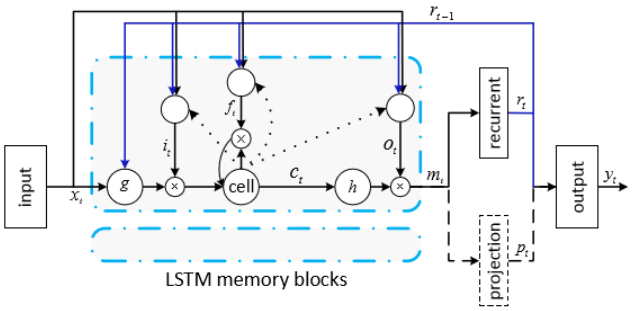

Figure 9: LSTM模型, 图来自[17]

4.2.3 GRU

GRU模型是Cho、van Merrienboer、Bahdanau和Bengio在2014年提出的。它是LSTM的一个变体，其性能可与LSTM 相媲美，但其设计显着减少了要估算的参数数量[14]。在整体性能方面，LSTM的表现优于GRU，准确率上表现的并不明显，而且GRU的计算效率更高。因此，我们常使用GRU来构建更深层的架构。

4.3 TDNN模型

TDNN在1989年被Hinton等人提出，用于音素识别。在近期才有研究者将其引入到语种识别领域，其取得的结果要优于LSTM。它的两个明显的特征是动态适应时域特征变化和参数较少，传统的深度神经网络的输入层与隐含层一一连接，TDNN改善了这一点，即隐含层的特征不仅与当前时刻的输入有关，而且还与未来时刻的输入有关。该模型具有以下优点：网络结构多层，每层对特征有较强的抽象能力；有能力表达语音特征在时间上的关系；具有时间不变性；学习过程中不要求对所学的标记进行精确的时间定位以及通过共享权值，方便学习。

4.4 CNN模型

由Maryam Najafian，Sameer Khurana等人提出CNN进行语种识别[12] ，Yu-Wen Lo, Yih-Liang Shen等人提出了一种嵌入式NN用于语音处理的生成听觉模型[18] 进行说话人识别。这些都说明了CNN在语音分类问题上的成功。

Yu-Wen Lo, Yih-Liang Shen等人提出CNN模型可分为两个阶段[18] ，如Figure 10所示，第一阶段是由一维卷积模拟的耳蜗过滤，第二阶段是由二维卷积模拟皮质过滤。

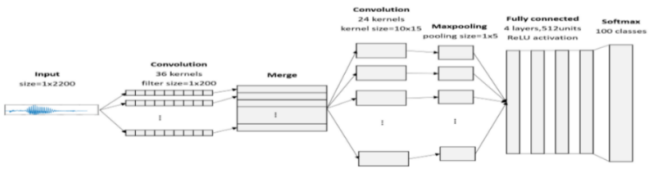

Fig. 2. Architecture of the proposed NN for speech processing on discriminative tasks.

Figure 10: CNN应用于语种识别系统, 图来自[18]

最近，针对输入语音长短不一问题，研究者们在神经网络体系结构中

采用时间平均pooling层（TAP）的方法[19] [20]，利用TAP的优点，神经网络能够训练具有持续时间的输入段。基于这种方法，研究者们又提出了CNN-LDE系统[21]，该系统中将CNN-TAP系统中平均pooling层用LDE层替换, 与简单的TAP不同，它依赖于可学习的字典，LDE 与TAP相比具有优越性和互补性。

Masschusetts等人还提出了一种端到端的方言辨识，利用卷积神经网络可直接将原始波形直接映射相应的方言[23]。

4.5 Attention模型

近些年关注机制被大量应用于自然语言处理中，基于大脑注意力模型原理被提出，Geng, Wang等人随后在[16]中第一次提出了基于注意力机制的递归神经网络结构，用以实现端到端语种识别的话语水平分类。这种语种识别模块受机器翻译的启发，与其他基于注意力机制的模型类似，将LSTM RNN用在编码输入序列的长跨度连接。但是基于注意力机制的序列到标签结构中的模型训练与推理和之前应用于自然语言处理中的序列到序列结构中的模型不一致。[16]提出通过查找表操作提供的注意力机制向量参与编码得到高级特征，然后从中选择输入序列中的关键特征代表帧级输入。根据是否对所有源框架或仅在少数源框架上“注意”，[16]中还开发了两种注意方法：软注意和硬注意方法。流程图如Figure 11 所示，实验结果与RNN相比较有了明显的提升。

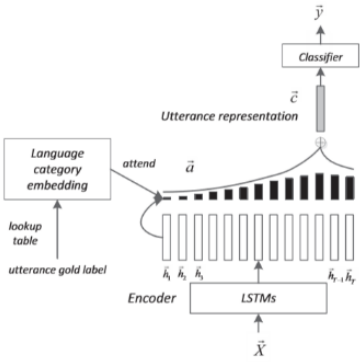

Figure 1: *The Architecture of attention-based recurrent neural network.*

Figure 11: Attention-based RNN LID系统框架, 图来自[25]

4.6 PTN模型

由Zhiyuan Tang等人提出的PTN模型[17]是一种以音素判别DNN产生的phonetic特征作为输入，而不是原始声学特征的LSTM-RNN LID系统。这个新模型类似于传统的phonetic LID方法，但这里的phonetic知识更丰富，它通过帧级判别训练可以学习短时语音信息与涉及所有作用的音素信息。与传统的Based-token方法有所不同，首先，PTN 方法中的语音信息是帧级的，而在传统的Based-token的方法中，该信息是单元级的。因此，PTN方法可以在更短的时间分辨率下表示语音属性。其次，传统的based-token的方法将语音信息表示为源自音素识别的序列，而PTN 方法将语音信息表示为涉及由所有音素的信息的特征向量，因此表示更详细的语音信息。最后，传统的基于token 方法的后端模型是基于离散token的n-gram LM并且使用最大似然（ML）标准训练，而PTN方法的后端模型是RNN。总之，PTN利用DNN phonetic特征和强大的LSTM模型来获取区分语种信息属性。PTN方法令许多LID研究人员重新认识到语音时间信息在语种识别中价值非凡。流程图如Figure 12所示。

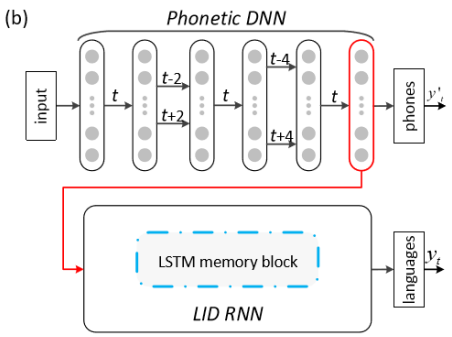

Figure 12: PTN-LID系统框架, 图来自[17]

基于分层思想模型的提出[22]，也给语种识别提供了一个新的发展方向。文献[22]论证了基于分层框架的语言模型能够比非分层方法更好地拒绝未知语言。

5 小结

本节介绍了语种识别的若干方法。总体来说，这些方法可以分为基于音素识别器的语种识别、基于底层声学特征的语种识别和基于深度学习的语种识别。基于音素识别器的语种识别，典型的如PRLM，将输入语音通过一个音素识别

器转化为token级别的音素，再将此特征进行N-gram单元进行统计，最后根据这些统计特性建立每个语种的N-gram语言模型。此方法需要音素识别器，因此该方法的一个局限性是语音转录的语音数据必须是可用的，以便用于基于音素识别器的LID的前端。基于底层声学特征的语种识别利用声学单元的统计特性差异来对语种进行分类，其所需特征直接通过底层的谱参数得到，不需要音素识别器作为支撑，因此一直以来都是语种识别研究的热点。基于深度学习的语种识别是近年的研究重点，虽然在该方法中需要更多的数据，但此方法对语种特性的学习更加细致，对短时语音的性能也更好。

References

[1] Weicheng Cai, Zexin Cai, Xiang Zhang, Xiaoqi Wang, and Ming Li, "A novel learnable dictionary encoding layer for end-to-end language identification," *arXiv preprint arXiv:1804.00385*, 2018.

[2] William M Campbell, Douglas E Sturim, and Douglas A Reynolds, "Support vector machines using gmm supervectors for speaker verification," *IEEE Signal Process. Lett.*, vol. 13, no. 5, pp. 308–311, 2006.

[3] Yeshwant K Muthusamy, Etienne Barnard, and Ronald A Cole, "Automatic language identification: A review/tutorial," *IEEE Signal Processing Magazine*, vol. 11, no. 4, pp. 33–41, 1994.

[4] Yeshwant Kumar Muthusamy, "A segmental approach to automatic language identification," *IEEE Signal Processing Magazine*, 1993.

[5] Marc A Zissman and Elliot Singer, "Automatic language identification of telephone speech messages using phoneme recognition and n-gram modeling," in *icassp*. IEEE, 1994, pp. 305–308.

[6] Marc A Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Transactions on speech and audio processing*, vol. 4, no. 1, pp. 31, 1996.

[7] Corinna Cortes and Vladimir Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[8] Pedro A Torres-Carrasquillo, Elliot Singer, Mary A Kohler, Richard J Greene, Douglas A Reynolds, and John R Deller Jr, "Approaches to language identification using gaussian mixture models and shifted delta cepstral features," in *Seventh International Conference on Spoken Language Processing*, 2002.

[9] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[10] Colin Raffel and Daniel PW Ellis, "Feed-forward networks with attention can solve some long-term memory problems," *arXiv preprint arXiv:1512.08756*, 2015.

[11] Geoffrey E Hinton and Ruslan R Salakhutdinov, "Reducing the dimensionality of data with neural networks," *science*, vol. 313, no. 5786, pp. 504–507, 2006.

[12] KV Mounika, Sivanand Achanta, HR Lakshmi, Suryakanth V Gangashetty, and Anil Kumar Vuppala, "An investigation of deep neural network architectures for language recognition in indian languages.," in *INTERSPEECH*, 2016, pp. 2930–2933.

[13] Maryam Najafian, Sameer Khurana, Suwon Shan, Ahmed Ali, and James Glass, "Exploiting convolutional neural networks for phonotactic based dialect identification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5174–5178.

[14] Alicia Lozano-Diez, Oldrich Plchot, Pavel Matejka, and Joaquin Gonzalez-Rodriguez, "Dnn based embeddings for language recognition," in *Proceedings of ICASSP*, 2018.

[15] Trung Ngo Trong, Ville Hautamäki, and Kong Aik Lee, "Deep language: a comprehensive deep learning approach to end-to-end language recognition," in *Odyssey: the Speaker and Language Recognition Workshop*, 2016.

[16] Abualsoud Hanani, Aziz Qaroush, and Stephen Taylor, "Classifying asr transcriptions according to arabic dialect," in *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)*, 2016, pp. 126–134.

[17] Wang Geng, Wenfu Wang, Yuanyuan Zhao, Xinyuan Cai, Bo Xu, et al., "End-to-end language identification using attention-based recurrent neural networks," in *INTERSPEECH*, 2016, pp. 2944–2948.

[18] Zhiyuan Tang, Dong Wang, Yixiang Chen, Lantian Li, and Andrew Abel, "Phonetic temporal neural model for language identification," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 134–144, 2018.

[19] Yu-Wen Lo, Yih-Liang Shen, Yuan-Fu Liao, and Tai-Shih Chi, "A generative auditory model embedded neural network for speech processing," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5179–5183.

[20] Chao Li, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, and Zhenyao Zhu, "Deep speaker: an end-to-end neural speaker embedding system," *arXiv preprint arXiv:1705.02304*, 2017.

[21] David Snyder, Pegah Ghahremani, Daniel Povey, Daniel Garcia-Romero, Yishay Carmiel, and Sanjeev Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 165–170.

[22] Saad Irtza, Vidhyasaharan Sethu, Eliathamby Ambikairajah, and Haizhou Li, "End-to-end hierarchical language identification system," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5199–5203.

[23] Suwon Shon, Ahmed Ali, and James Glass, "Convolutional neural networks and language embeddings for end-to-end dialect recognition," *arXiv preprint arXiv:1803.04567*, 2018.

[24] Ignacio Lopez-Moreno, Javier Gonzalez-Dominguez, Oldrich Plchot, David Martinez, Joaquin Gonzalez-Rodriguez, and Pedro Moreno, "Automatic language identification using deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 5337–5341.