深度神经网络(DNN)

洪青阳 副教授

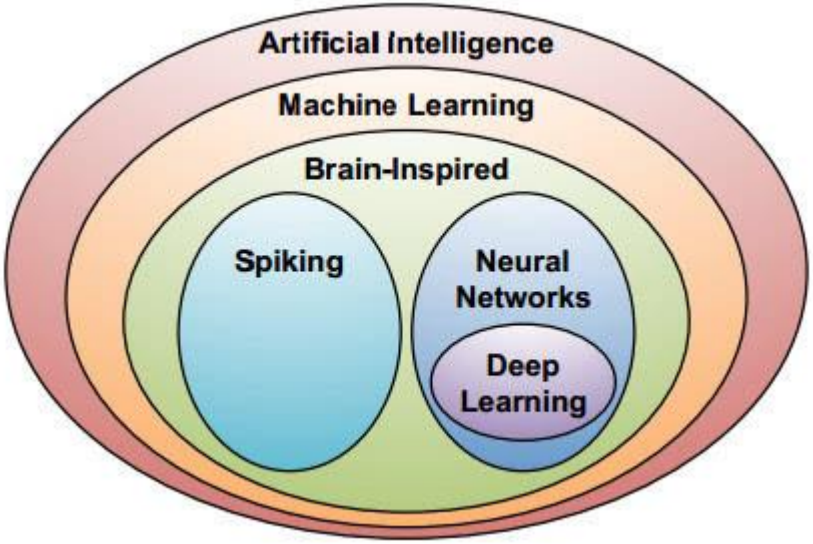
厦门大学信息科学与技术学院
qyhong@xmu.edu.cn

要点

- 深度学习
- 深度神经网络(DNN)
- DNN-HMM
- CNN
- 时延神经网络(TDNN)
- CTC
- Chain
- Kaldi工具

深度学习

- 近20年来，人工智能(AI)领域最具突破性的技术；
- 深度神经网络(DNN)在计算机视觉、语音识别、自然语言处理有广泛应用，并拓展到自动驾驶、疾病诊断、游戏AI等领域。



“深度”

- 2006年，Hinton利用预训练方法缓解了神经网络局部最优解问题，将隐含层推动到了7层，神经网络真正意义上有了“深度”，由此揭开了深度学习的热潮。

- 这里的“深度”并没有固定的定义——在语音识别中4层网络就能够被认为是“较深的”，Google产品voice search中采用的深度神经网络为4-5层，百度语音搜索采用的深度神经网络多达9层。现在已超过100层(ResNet)。

深度学习与语音识别

- 现有的语音识别技术在应用过程中仍然存在着鲁棒性差、识别率低等突出问题，这主要是因为传统的语音建模工具无法准确地描述语音内部复杂的结构，且表征和建模能力不强。

- 美国微软公司率先成功地将深度学习方法应用于自己的语音识别系统当中，它能够将单词错误率相较之前的最优方法降低约30%。因此，深度学习被认为是继隐马尔可夫(HMM)模型之后，语音识别领域的又一次重大突破。

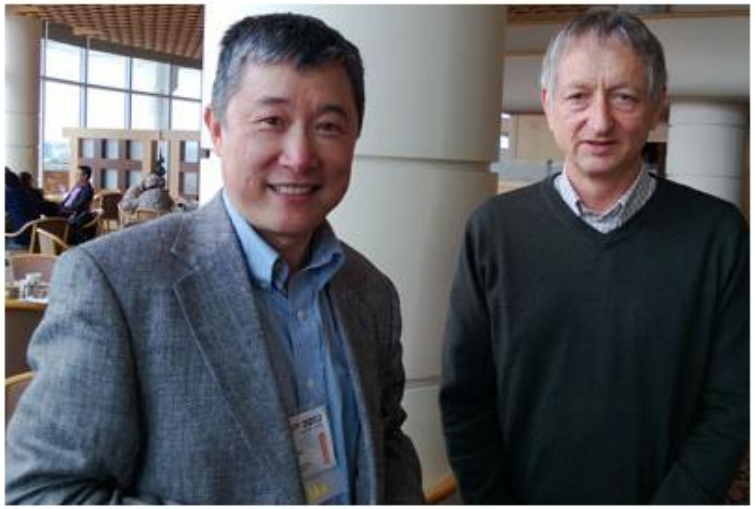
Enabling Cross-Lingual Conversations in Real Time

Microsoft Research
May 27, 2014 5:58 PM PT

The success of the team's progress to date was on display May 27, in a talk by Microsoft CEO Satya Nadella in Rancho Palos Verdes, Calif., during the Code Conference. During Nadella's conversation with Kara Swisher and Walt Mossberg of the Re/code tech website relating to a new

The path to the Skype Translator gained momentum with an encounter in the autumn of 2010. Seide and colleague Kit Thambiratnam had developed a system they called The Translating! Telephone for live speech-to-text and speech-to-speech trans calls.

e asked
tage. Pall, the
ident of
first time
app, with Pall
rman-
Diana



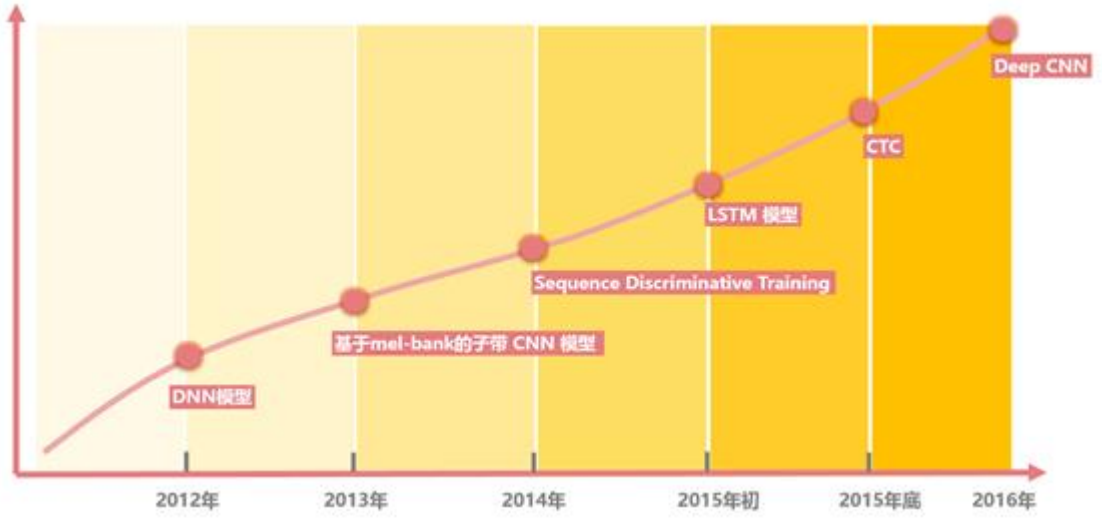
Li Deng (*left*) and Geoff Hinton.

A core development that enables Skype translation came from Redmond researcher Li Deng. He invited Geoff Hinton, a professor at the University of Toronto, to visit Redmond in 2009 to work on new neural-network learning methods, based on a couple of seminal papers from Hinton and his collaborators in 2006 that had brought new

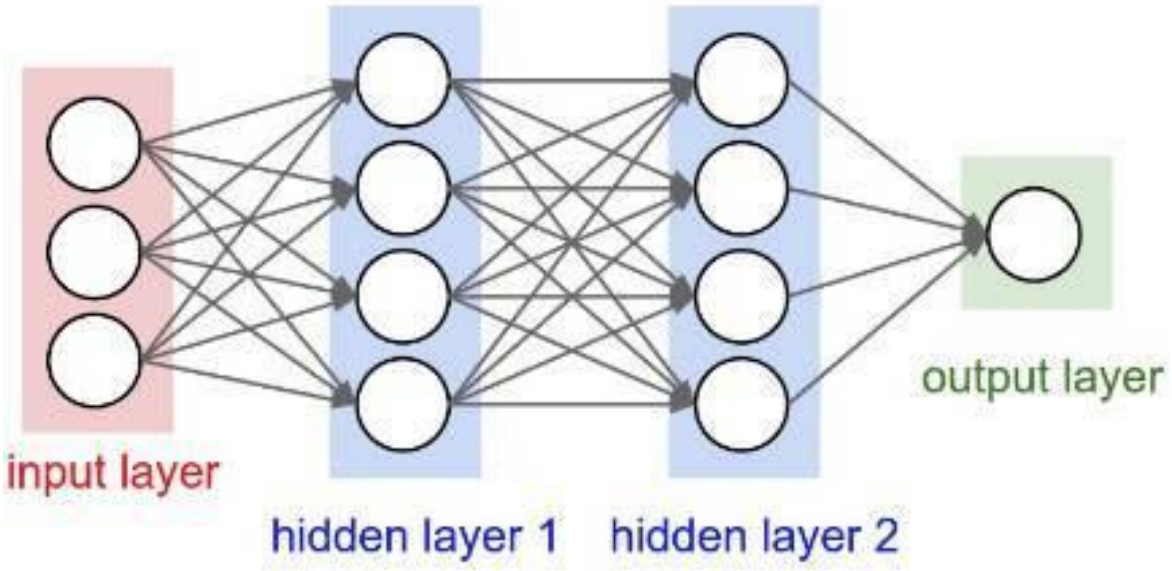
深度学习在工业界的应用

深度学习在工业界的应用

- 百度语音识别技术的迭代：
  - 2012年采用DNN模型，
  - 2013年改进为基于mel-bank的子带CNN模型，
  - 2014年加入区分度模型，
  - 2015年推出了LSTM和CTC模型，技术的迭代让语音识别准确率不断提升，已经超过96%，
  - 2016年，采用Deep CNN模型，准确率又提升了10% 。



深度神经网络(DNN)



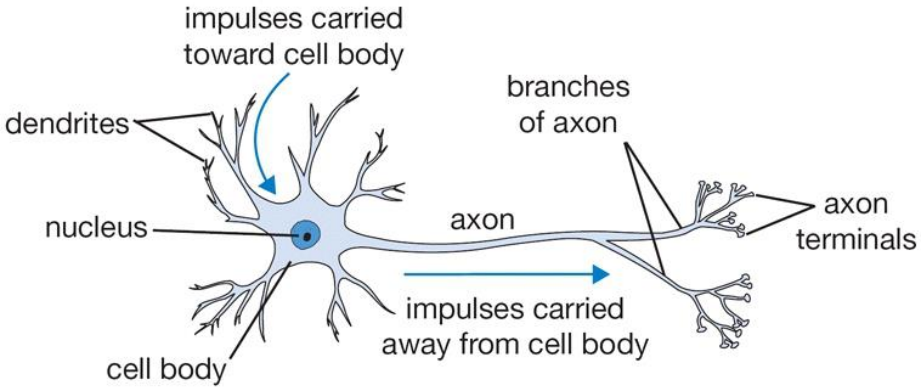
From: http://cs231n.github.io/neural-networks-1

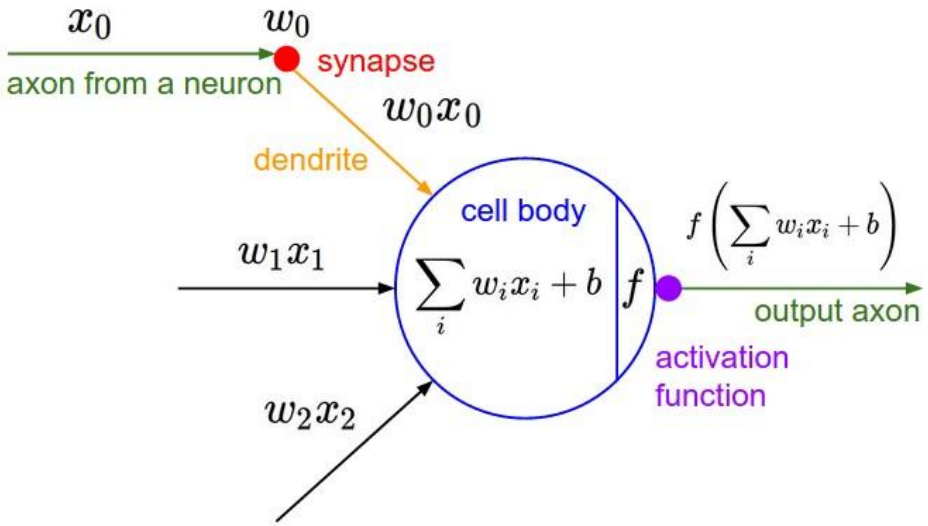
神经元

- 神经元是大脑的主要计算单元。人类大脑平均有860亿个神经元。神经元相互连接，通过树突接受其他神经元的信号，对这些信号进行计算之后，通过轴突将信号传递给下一个神经元。据估计，人类大脑平均有1014-1015个突触。

- 突触的一个关键特性是它可以缩放通过它的信号大小，这个比例因子可以被称为权重(weight)。普遍认为，大脑学习的方式是通过改变突触的权重实现的。因此，不同的权重导致对输入产生不同的响应。

神经元及数学模型

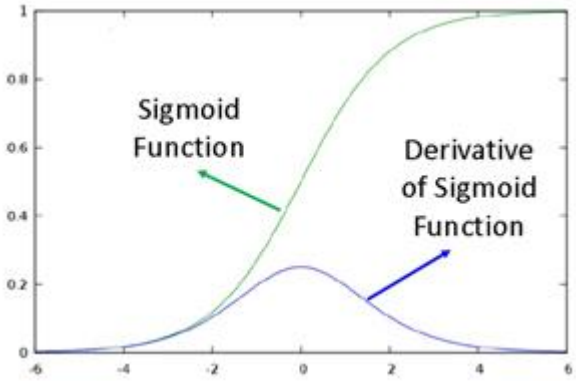




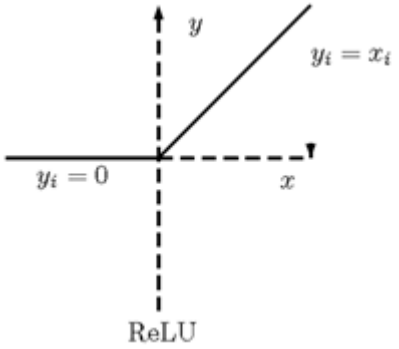
From: http://cs231n.github.io/neural-networks-1

激活函数

- Sigmoid: $\sigma(x) = 1/(1+\exp(-x))$
  - $\sigma'(x) = \sigma(x)(1-\sigma(x))$



- ReLU: $y = \max(0, x)$



- Softmax: $f(x_i) = e^{x_i} / \sum_{k=1}^{K} e^{x_k}$
  - $f'(x_i) = f(x_i)\big(I_{ij} - f(x_i)\big)$，$I_{ij}$表示单位矩阵的元素。

前向传播(FeedForward)

x_0为输入向量

$$y_i = W_i * x_{i-1} + b_i$$
$$x_i = f(y_i)$$

f通常是Sigmoid或ReLU函数

$$Sigmoid(y) = \frac{1}{1 + exp(-y)}$$
$$ReLU(y) = y > 0 ? y : 0$$

分类任务输出层通常为

$$Softmax(y_i) = \frac{exp(y_i - y_{max})}{\sum exp\,(y_i - y_{max})}$$

代表输入向量属于各个类的概率。

反向传播(BackForward)

- 训练准则
- 训练算法

训练准则—最小化目标函数

- 交叉熵(Cross Entropy, CE)

$$C = -\sum_{x} p(x) lnq(x)$$

- 均方误差(Mean Square Error, MSE)

$$C = -\sum_{x} \big(p(x) - q(x)\big)^2$$

其中p是真实样本分布，q是估计的模型分布。

训练算法

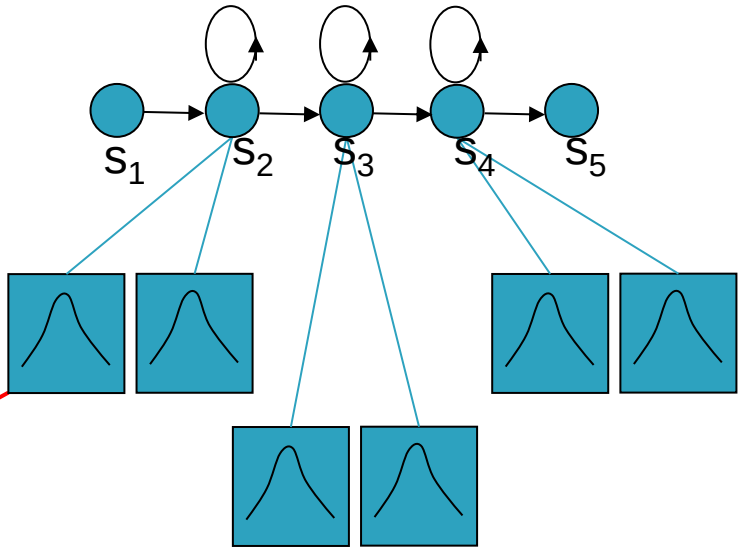
- 随机梯度下降(SGD)算法

$$W_{t+1} \leftarrow W_t - \varepsilon \Delta W_t$$
$$b_{t+1} \leftarrow b_t - \varepsilon \Delta b_t$$

其中ε是学习率。

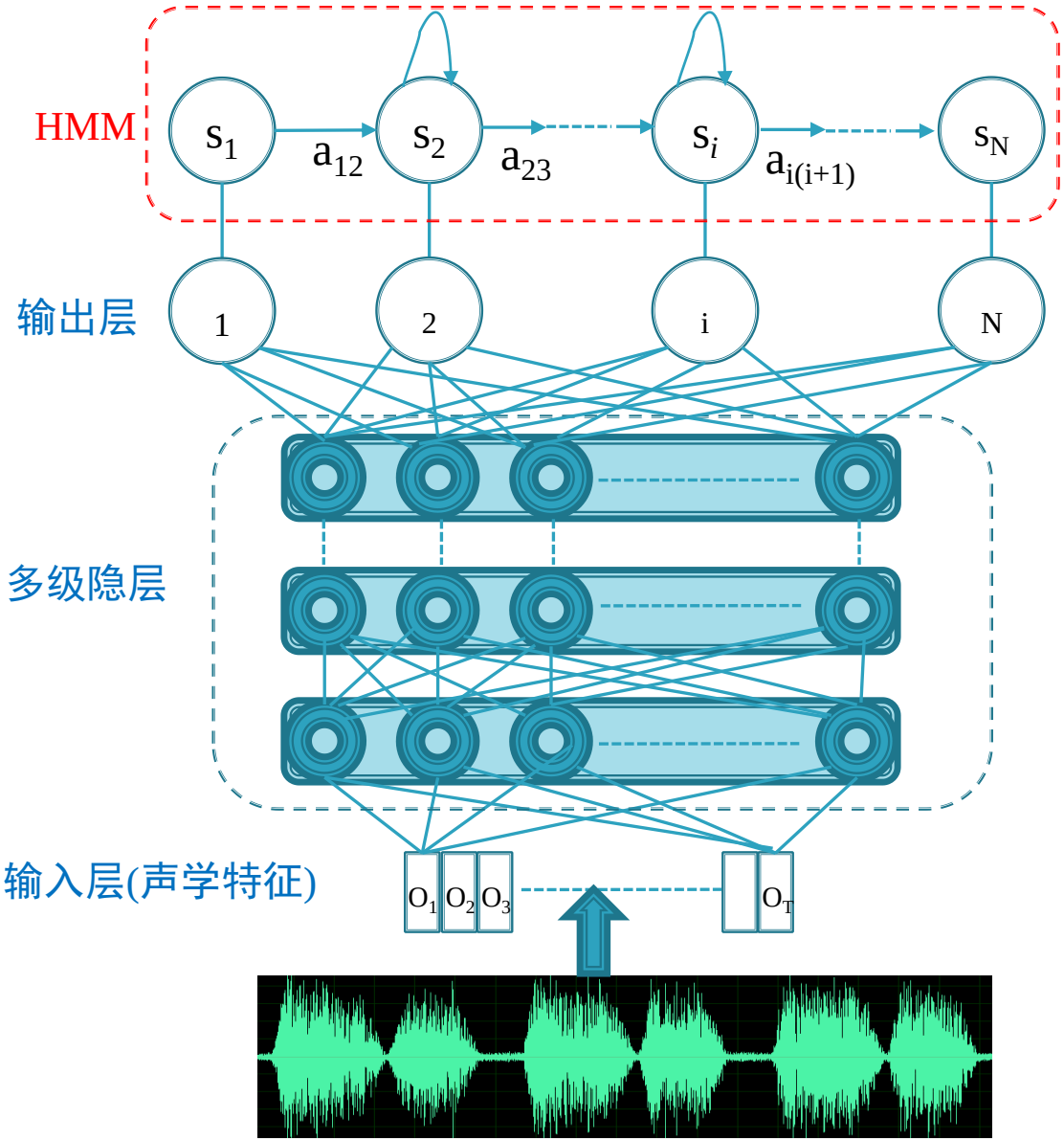
- 本质上是一个串行的算法，收敛相对较慢。

DNN-HMM



- GMM被DNN替代；
- DNN的Layout输出节点与HMM的状态节点一一对应，因此可通过DNN的输出得到每个状态针对观察值的概率；
- HMM的转移概率和初始状态概率保持不变。

DNN-HMM



DNN-HMM的训练

Step1：利用GMM-HMM系统，生成训练数据帧与状态的对齐；

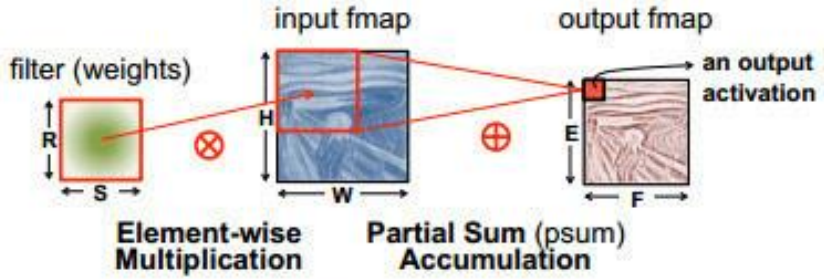
Step2：创建初始的DNN-HMM，包括输入层、输出层，开始只有一个隐藏层；

Step3：帧层次DNN预训练(pretrain)；

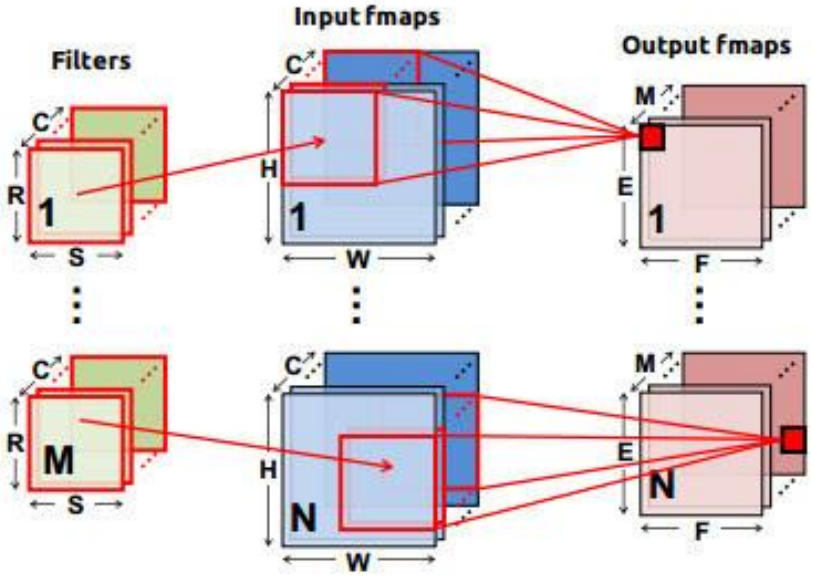
Step4：增加隐藏层，重新预训练；

Step5：达到需要的隐藏层数，预训练后再做优化训练(finetune)。

卷积神经网络(CNN)



(a) 2-D convolution in traditional image processing

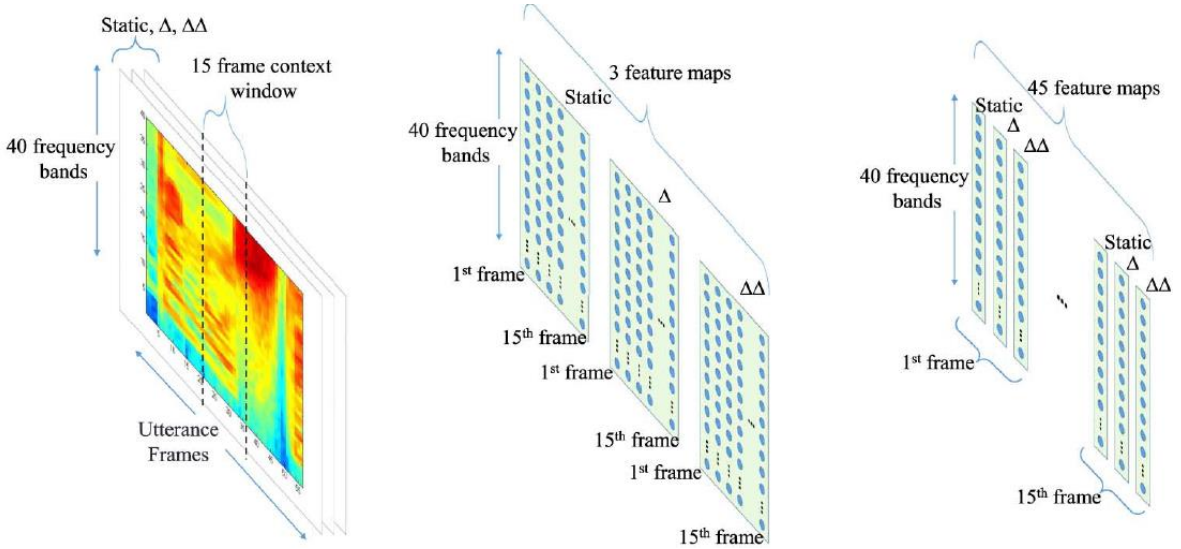


(b) High dimensional convolutions in CNNs

From: Vivienne Sze, Yu-Hsin Chen, Tien-Ju Yang, Joel S. Emer: Efficient Processing of Deep Neural Networks: A Tutorial and Survey. CoRR abs/1703.09039 (2017)

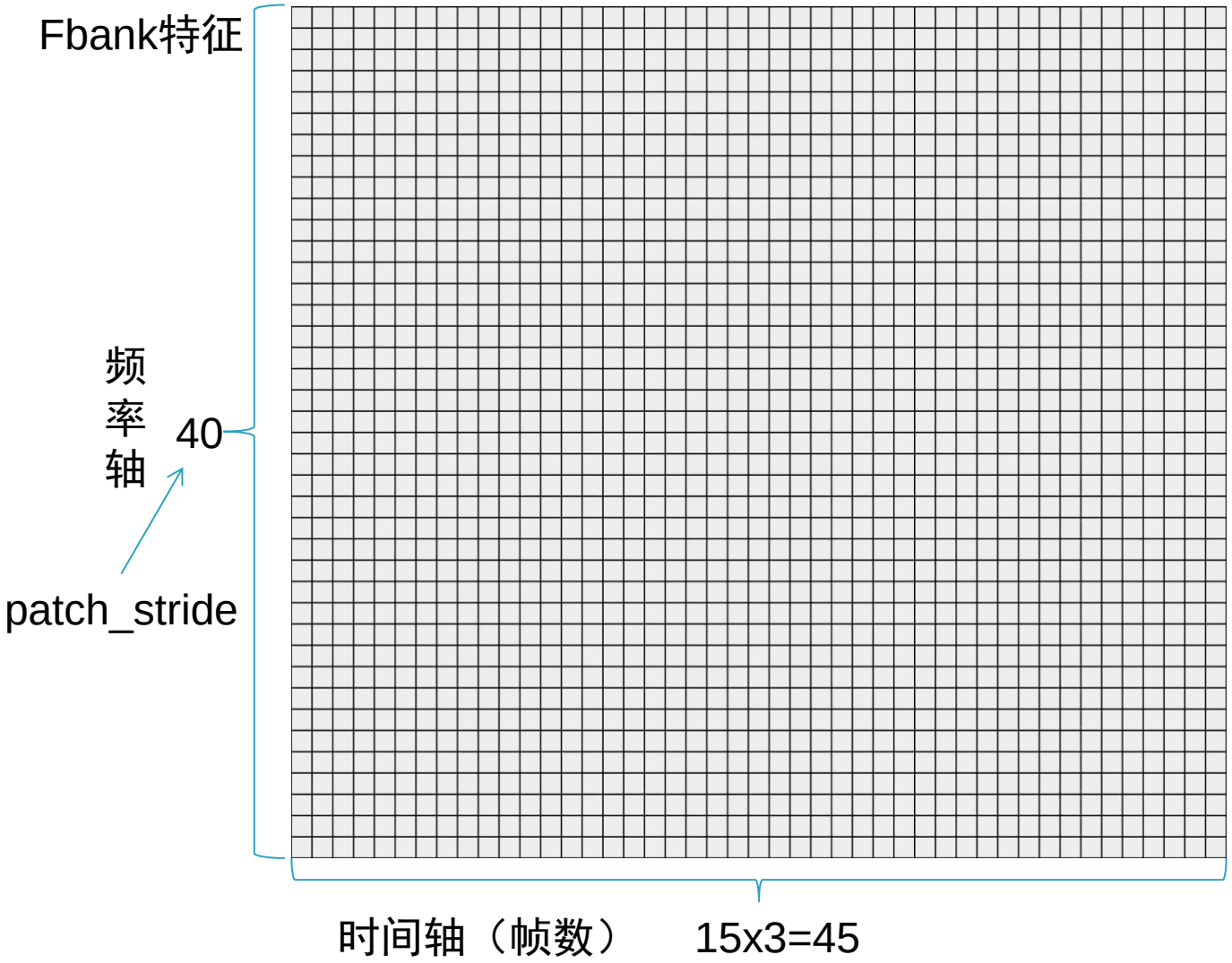
CNN的输入

- 特征维度：40 (Fbank)
- 一阶和二阶差分
- 连续15帧，加上差分，放大为45
- 则CNN的输入维度(input_dim)为：40*45



From: Ossama Abdel-Hamid, Abdel-rahman Mohamed, Hui Jiang, Li Deng, Gerald Penn, Dong Yu: Convolutional Neural Networks for Speech Recognition. IEEE/ACM Trans. Audio, Speech & Language Processing 22(10): 1533-1545 (2014)

输入特征

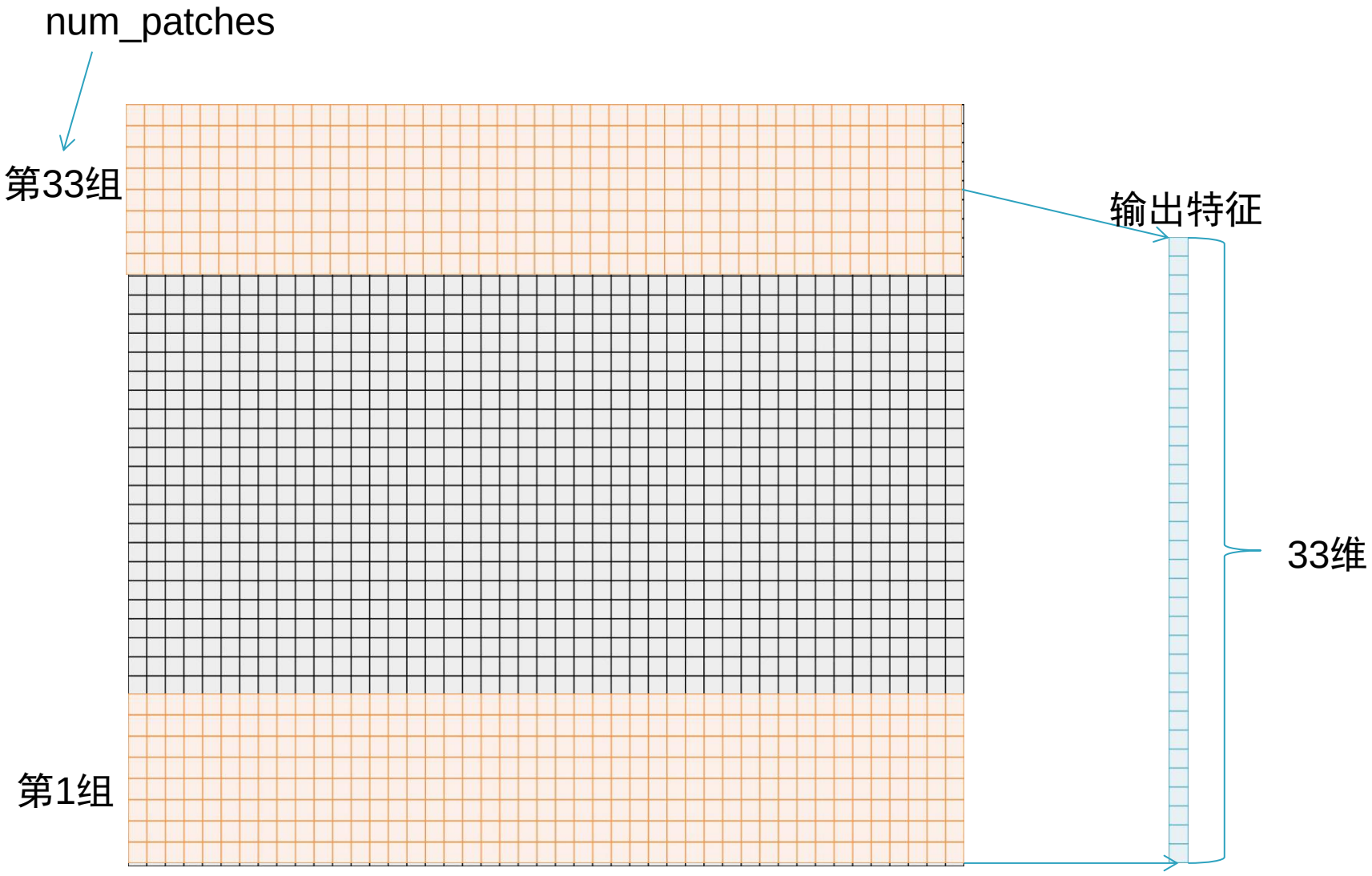


使用的filter

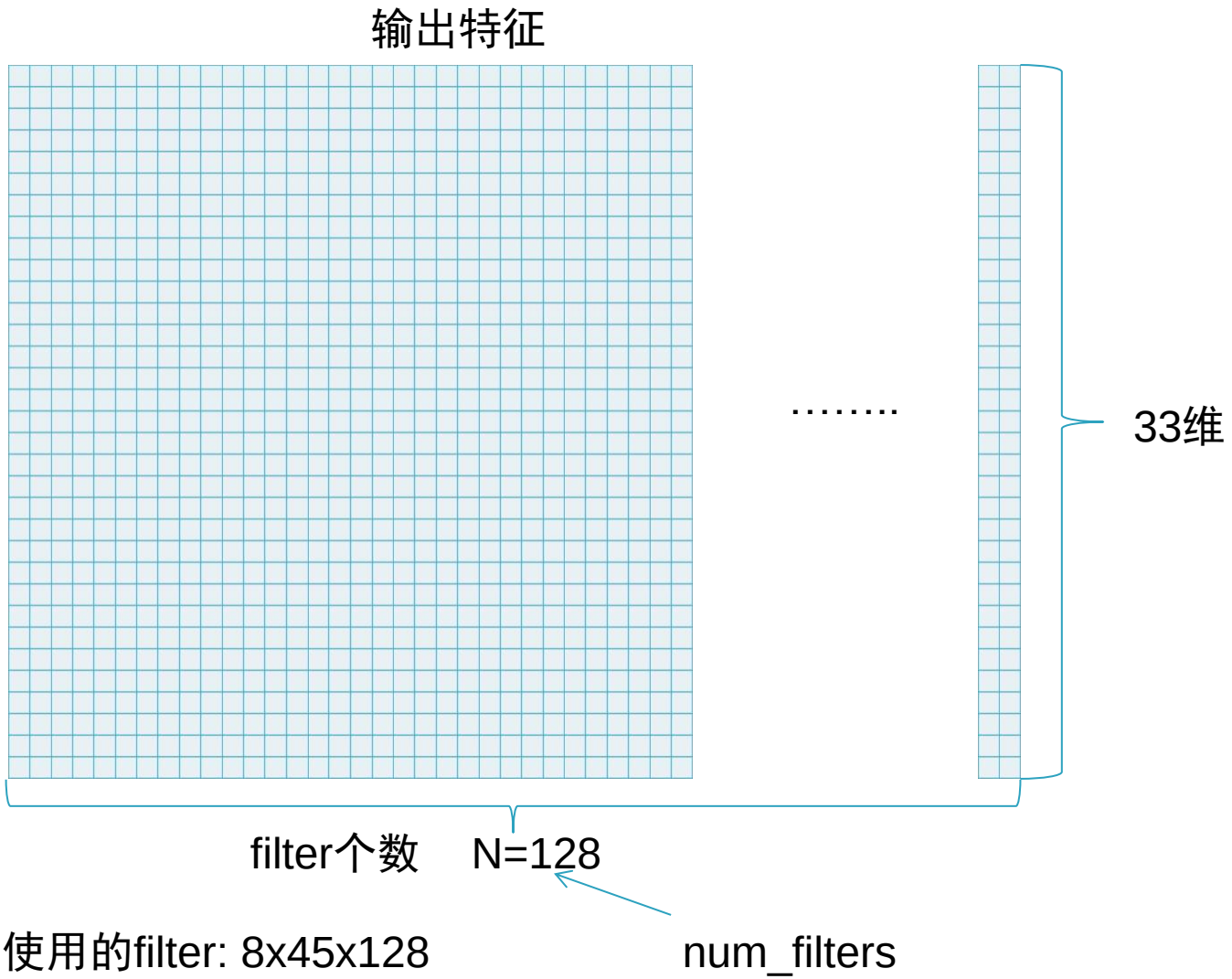
- 单个filter: 8*45

- 频率轴步进为1，特征维度40，则filter输出节点: (40-8)/1+1=33

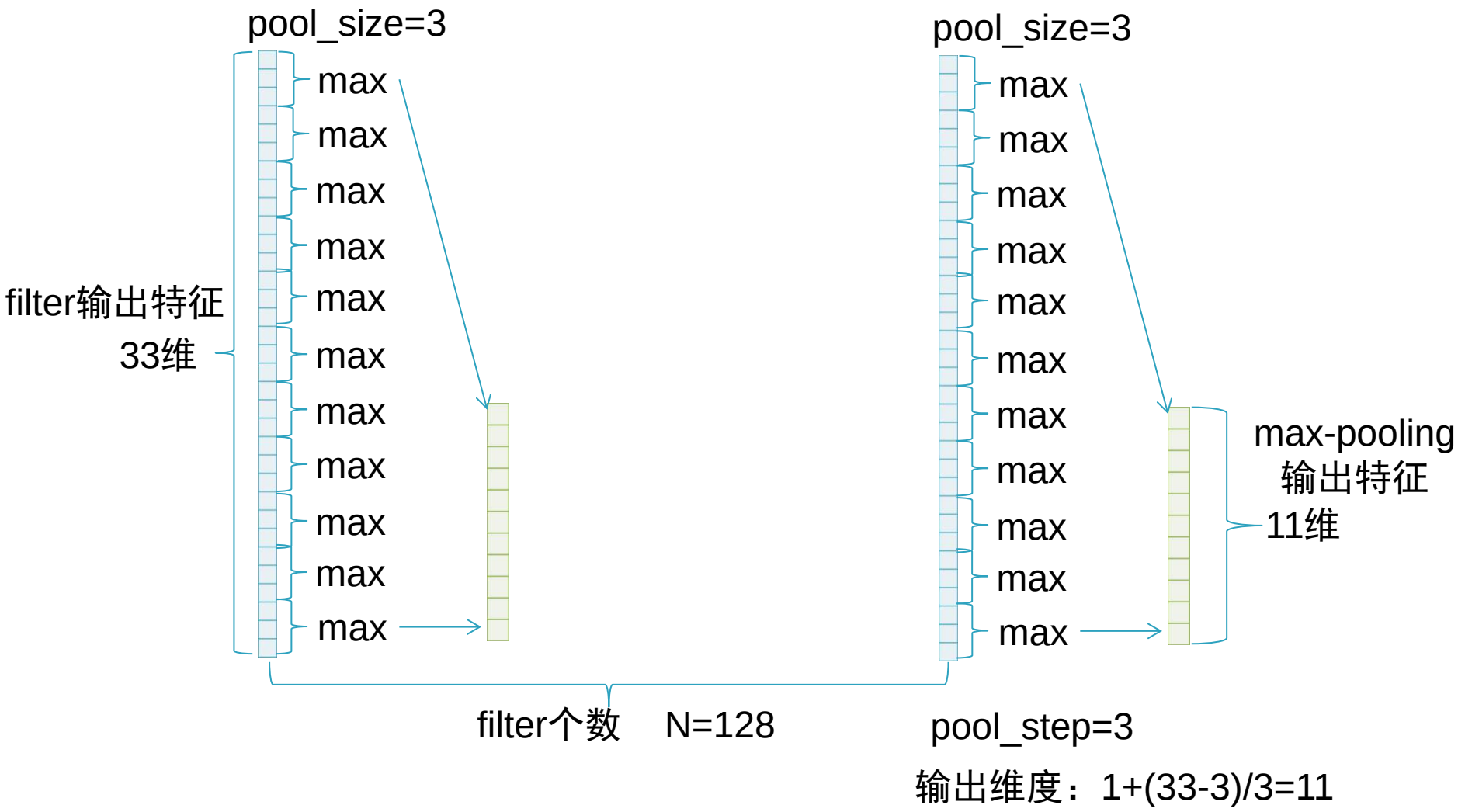
单个filter输出



N个filter输出



max-pooling输出



参数表示

- 假设输入40维的fbank，feat_raw_dim=40，包括一阶二阶差分，splice_num=15x3，delta_order=0：

  num_filters1=128
  num_filters2=256
  pool_size=3
  pool_step=3
  pool_type=MAX
  num_splice=45
  patch_step1=1
  patch_dim1=8
  patch_dim2=4

卷积层Conv1

patch_stride2
patch_stride2=feat_dim
一个滤波器参数：
filter_dim1=num_splice * patch_dim_=45*8=360
所有滤波器参数：
param1=num_filters1*filter_dim1=128*360=46080
cnn1输入为：
input1=splice_num*feat_raw_dim=45*40=1800
cnn1滤波器输出
num_patch1 = 1 + (feat_raw_dim - o.patch_dim1) / o.patch_step1 = 1 + (40 - 8) / 1 = 33
output1=num_filters1*num_patch1=128*33=4224

相当于输入是40*45的图片，使用8*45的filter只在频率维度上面进行step为1的移动，1个filter可以得到33*1的输出，128个filter可以得到output_dim为128*33的输出。

池化层max-pooling

pool个数
num_pool = 1 + (num_patch1 - o.pool_size) /
o.pool_step=1+(33-3)/3=11

pool_stride，conv1的filter的个数

pool_stride=o.num_filters1=128

output

o.num_filters1*num_pool=128*11=1408

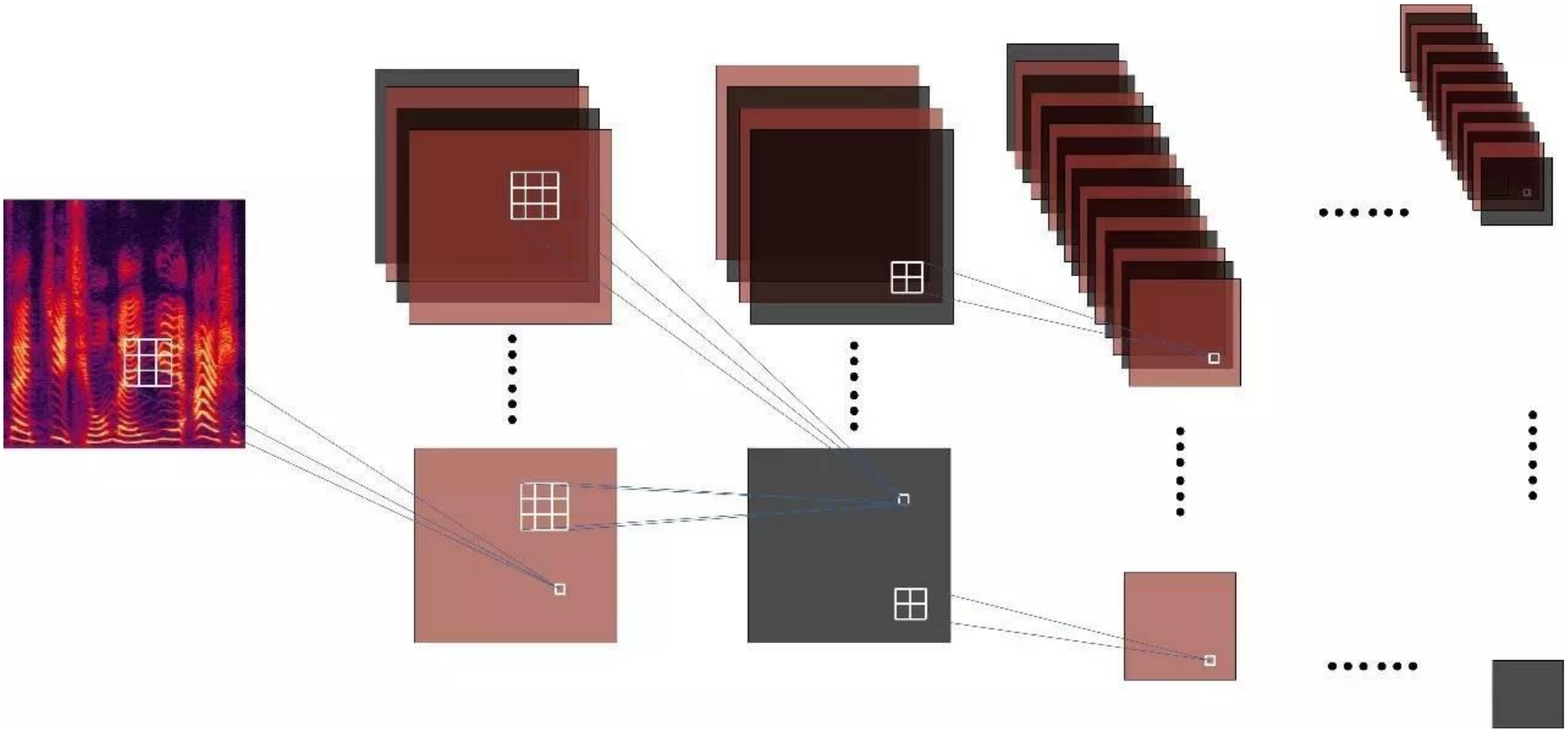
conv1每个滤波器的输出为33，在33这个维度上面，每3个数取一个最大值，得到11个数。128个滤波器就可以得到128*11的矩阵。

卷积层Conv2

输入为max-pooling的输出
input2=128*11
step和conv1一致
patch_step2 = o.patch_step1=1
stride和pooling的num_pool一致
patch_stride2 = num_pool=11
splice为filter1的个数
num_splice2=input2/patch_stride2=128
patch
num_patch2 = 1 + (num_pool - patch_dim2) / patch_step2=1+(11-4)/1=8
一个滤波器参数
filter_dim2=num_splice2 * patch_dim2_=128*8

其他跟conv1的计算方法一样。
相当于输入是11*128的图片，使用4*128的filter进行step为1的移动，得到8*1的输出，256个filter可以得到256*8=2048的输出。

Deep CNN



Deep CNN语音识别的建模过程

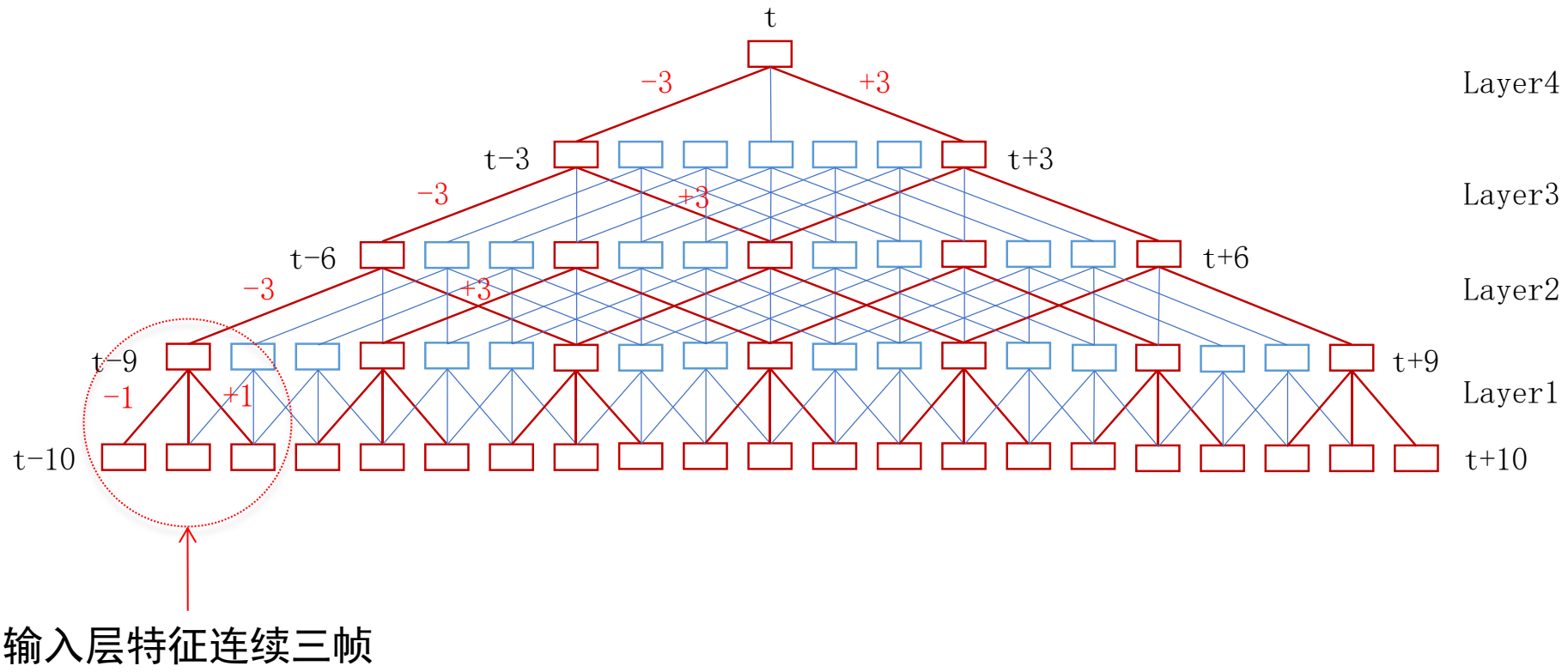
图片来源： https://blog.csdn.net/starzhou/article/details/53319472

时延神经网络(TDNN)

- TDNN最早用于音素识别[1]，是CNN的先驱。
- TDNN从底层到顶层，每层的输入都通过下层的上下文窗口获得。
- 能够描述上下层结点之间的时间关系。

[1] Alexander H. Waibel, Toshiyuki Hanazawa, Geoffrey E. Hinton, Kiyohiro Shikano, Kevin J. Lang: Phoneme recognition using time-delay neural networks. IEEE Trans. Acoustics, Speech, and Signal Processing 37(3): 328-339 (1989)

TDNN网络结构图

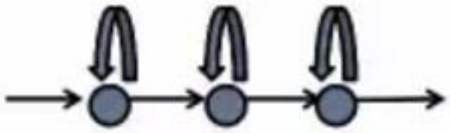

t
-3
+3
Layer4
t-3
t+3
-3
+3
Layer3
t-6
t+6
-3
+3
Layer2
t-9
t+9
-1
+1
Layer1
t-10
t+10
输入层特征连续三帧

CTC (Connectionist Temporal Classification)

- 假设一段长为T的序列的每个时间点的label有可能是L个不同的label中的一个，那么这段序列的label总共有L的T次方种可能；
- 在CTC模型中，一般把L个label扩展到L+1个label，多加一个空白字符的label，表示序列中的空白输出；
- CTC使得训练过程无需帧级别的标注，实现有效的“end-to-end”训练。

CTC

1. HMM + GMM
2. HMM + DNN

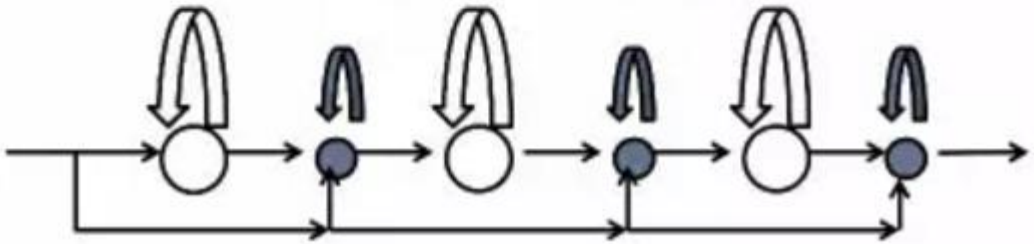


3. CTC :

a b c = blank a a b blank c c c blank

= blank a blank b b blank c blank

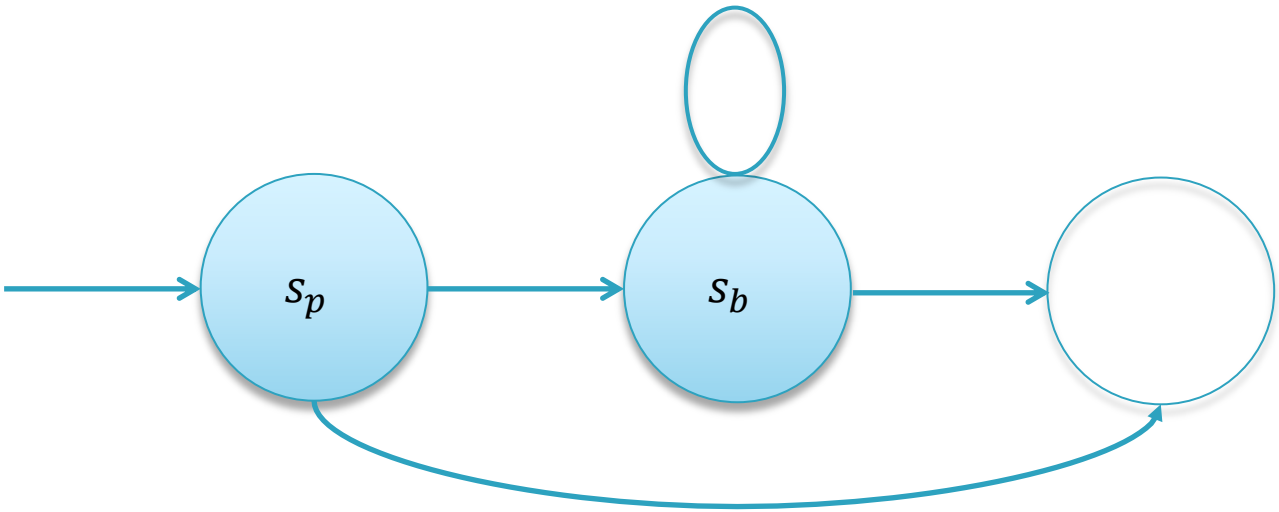
= blank a a a a blank b b b c c blank



CTC与HMM的差异

- 第一是模型结构差异，CTC引入了blank。
- 第二个CTC训练无需固定边界，自动的end to end优化模型参数。
- CTC 打破了隐马尔科夫的假设，把整个模型从静态分类变成了序列分类。

Chain模型



s_p—phone的状态，s_b—blank的状态

Chain模型

- 灵感来源于 CTC 训练方法；
- HMM 状态数从 3 或 5 降为 1，实际还有一个用于自旋可重复 0 次或多次的空白状态；
- 降低了帧率(间隔3帧，即帧移从10ms改为30ms)；
- 固定了 HMM转移概率；
- 相比传统的 TDNN 模型，Chain+TDNN 模型在解码速度和准确率上都有明显提升。

Kaldi工具

- Kaldi是一个开源的语音识别工具箱，是基于C++编写的，可以在Linux和Windows平台上编译。
- Kaldi 的特色：
  - 与文本无关的LVCSR系统；
  - 基于FST的训练和解码；
  - 最大似然训练；
  - 各种各样的线性和映射变换；
  - 有VTLN，SAT的脚本；
- Kaldi 的声学模型：
  - 支持标准的机器学习训练模型：
  - 线性变换如：LDA HLDA,MLLT/STC；
  - 说话人自适应:fMLLR,MLLR；
  - 支持GMM,SGMMs,DNN

Kaldi工具

- Kaldi 在学术界和工业界被广泛采用：MIT、CMU、微软、谷歌、阿里巴巴等都有在用 Kaldi 做研究工作。Kaldi 主干版本是由 Dan Povey 博士维护。

- 高效地保持更新：
  - 从GMM到DNN，
  - 支持 GPU 训练，
  - 从近场识别扩展到远场识别，
  - 效率和性能更高的 Chain Model，
  - 训练过程加入对抗学习。

Thank you!

Any questions?