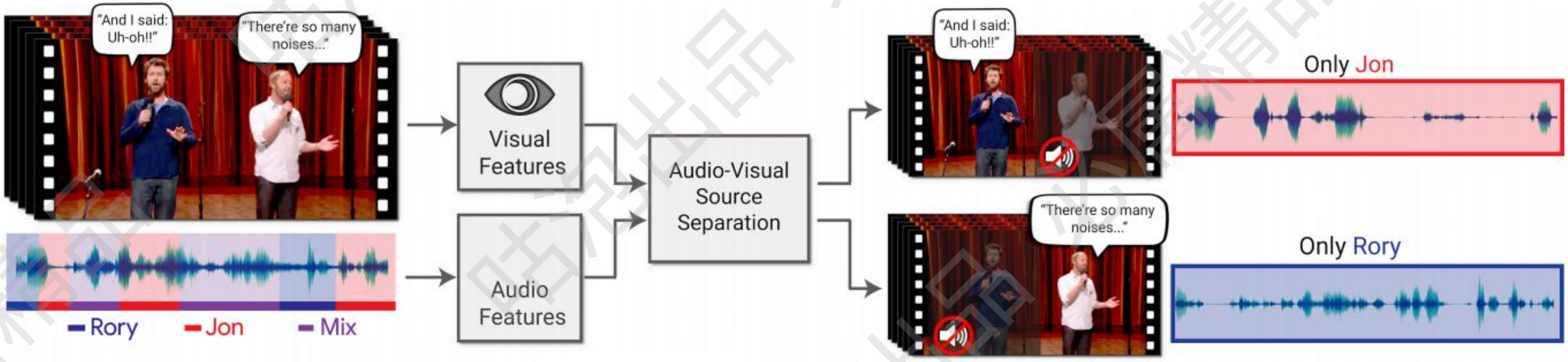
Separation

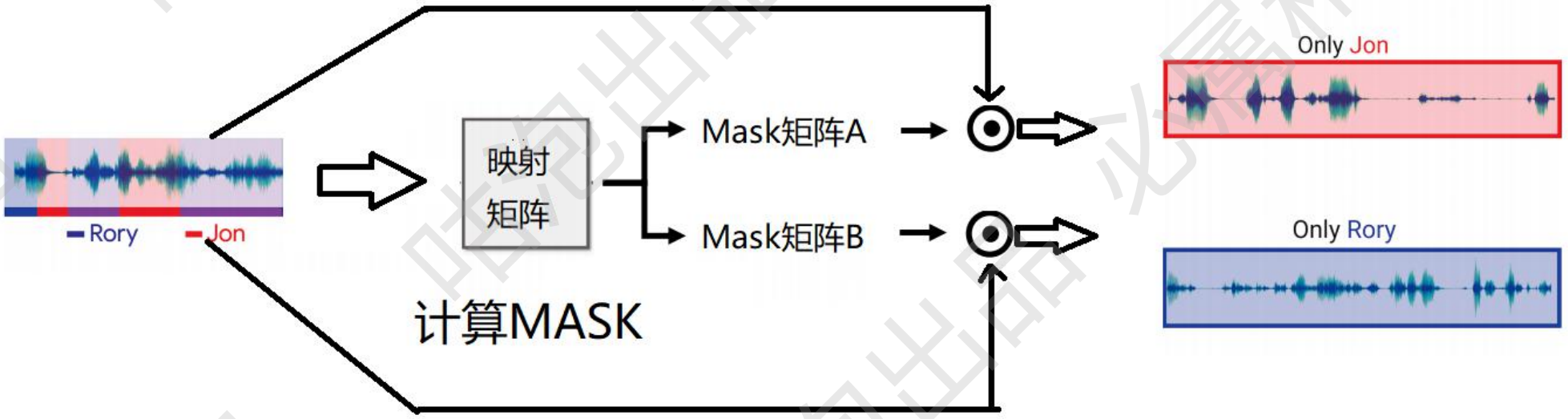
- 什么是语音分离呢?

- 输入为混合的声音，输出各个讲话者单独的声音:



Separation

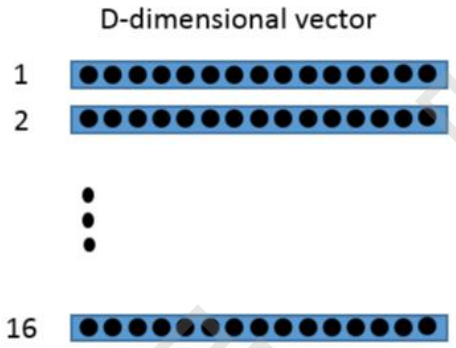
- 经典的Deep Clustering
  - 在混合的声音信号中，拿出来每一个人的声音不就好了！
  - 并不需要对信号做特殊的变换处理，想办法分离出来每一个



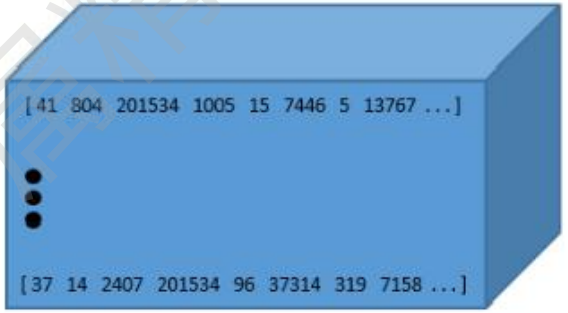
Separation

- 求解流程

- 先得到输入特征，例如16个采样点，每个为D维特征：
  （可以先假设输入的就是两个人的语音信号，其中0表示第一个人，1表示第二个人）



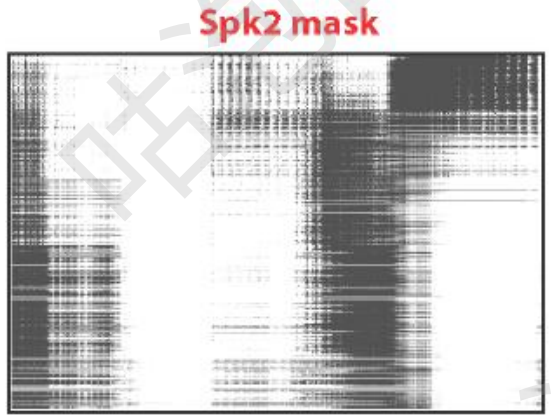
- 像文本数据一样，训练一个Embedding：
  （相当于每一个点都变成一个向量了，这样每一个点就可以当做一个样本了）

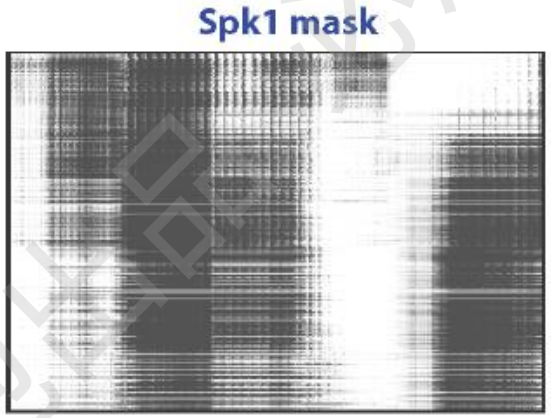


Separation

- 求解流程

  - 对得到的所有“样本”进行Kmeans聚类操作，可以得到其每一个的类别
  - 如果原始输入数据有两个人，K就等于2！
  - 把得到的label结果当做mask矩阵就可以啦，分离出来，两个mask就搞定了！



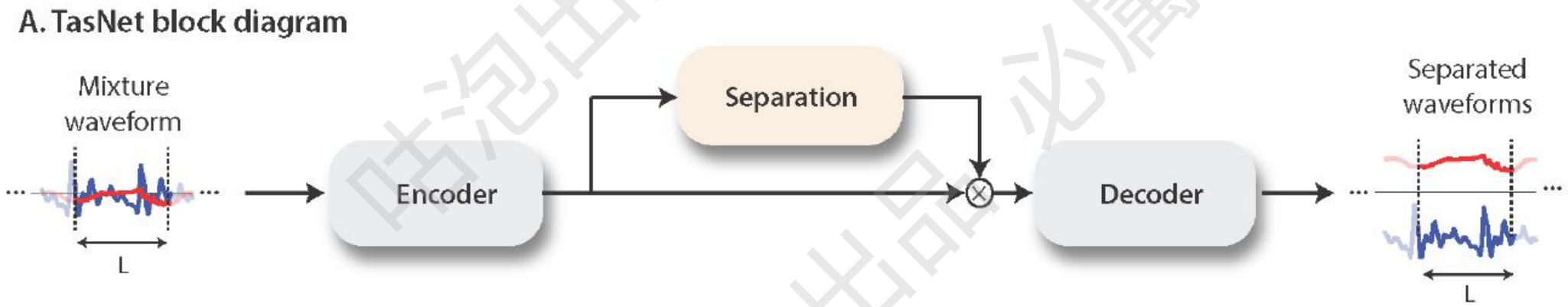


Separation

- Conv-TasNet

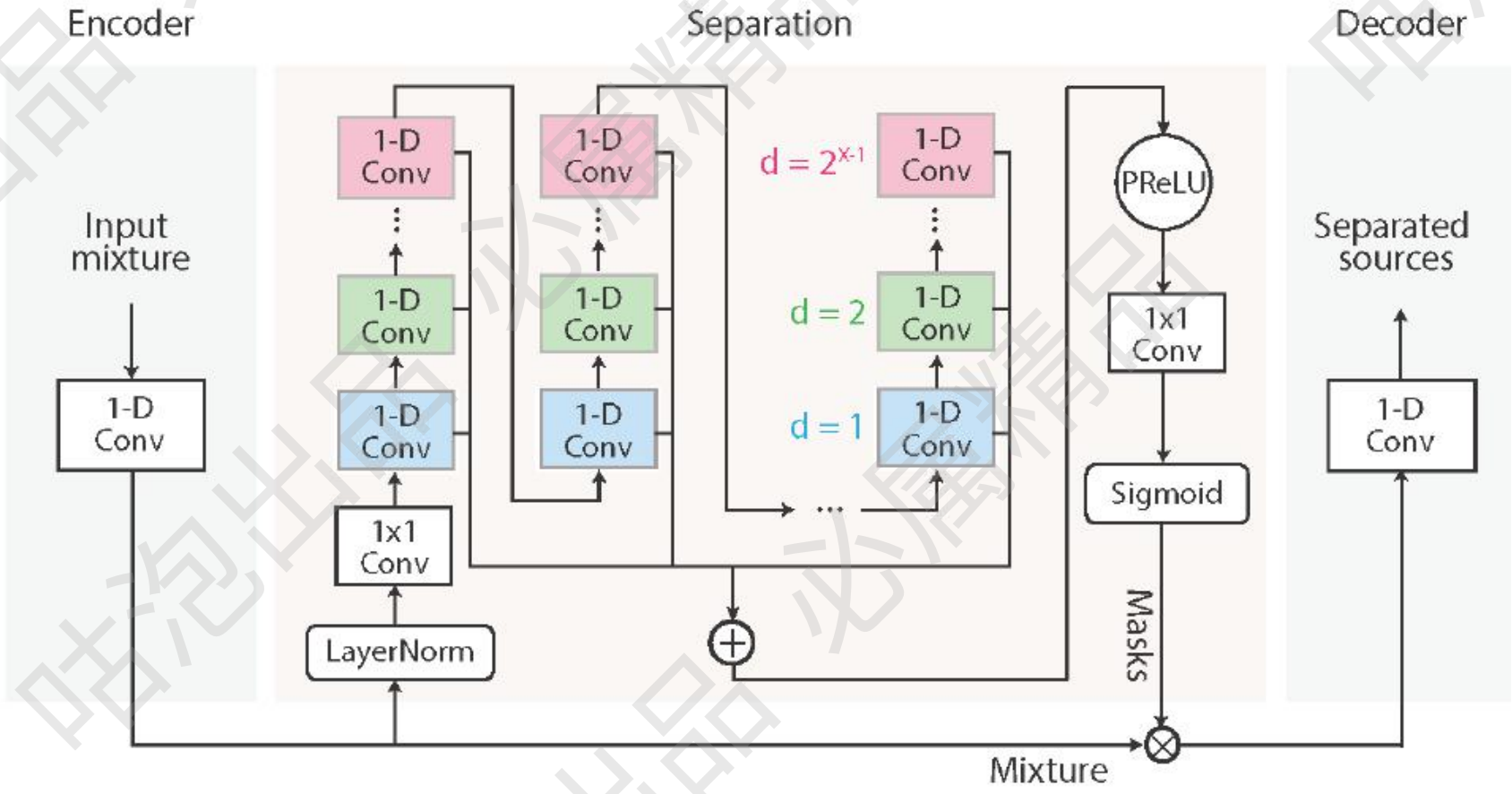
  - 可别整那么复杂了，又得提特征，又得embedding，还得聚类的。。。

  - 现在啥不流行个一条龙服务啊！可以当做是编码，解码的过程！



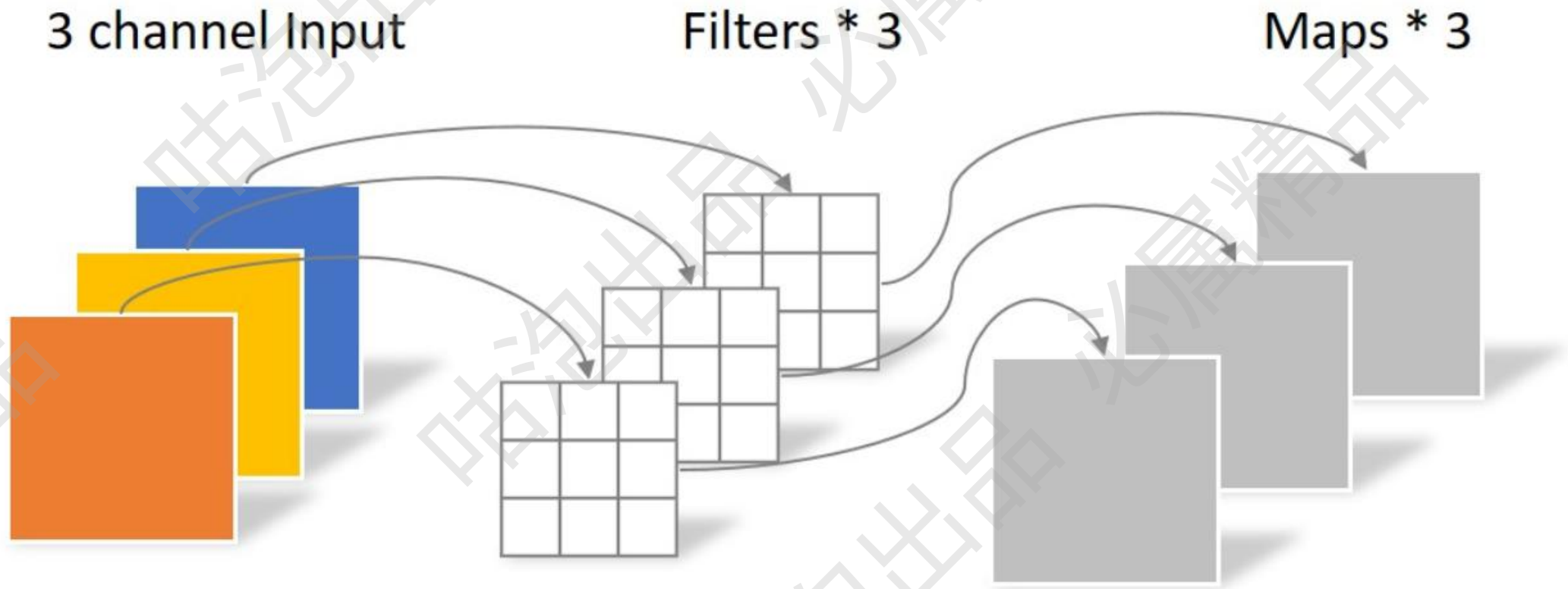
Separation

- 网络结构细节：
  - 全卷积
  - 更大的感受野
  - 纯语音输入输出
  - end to end



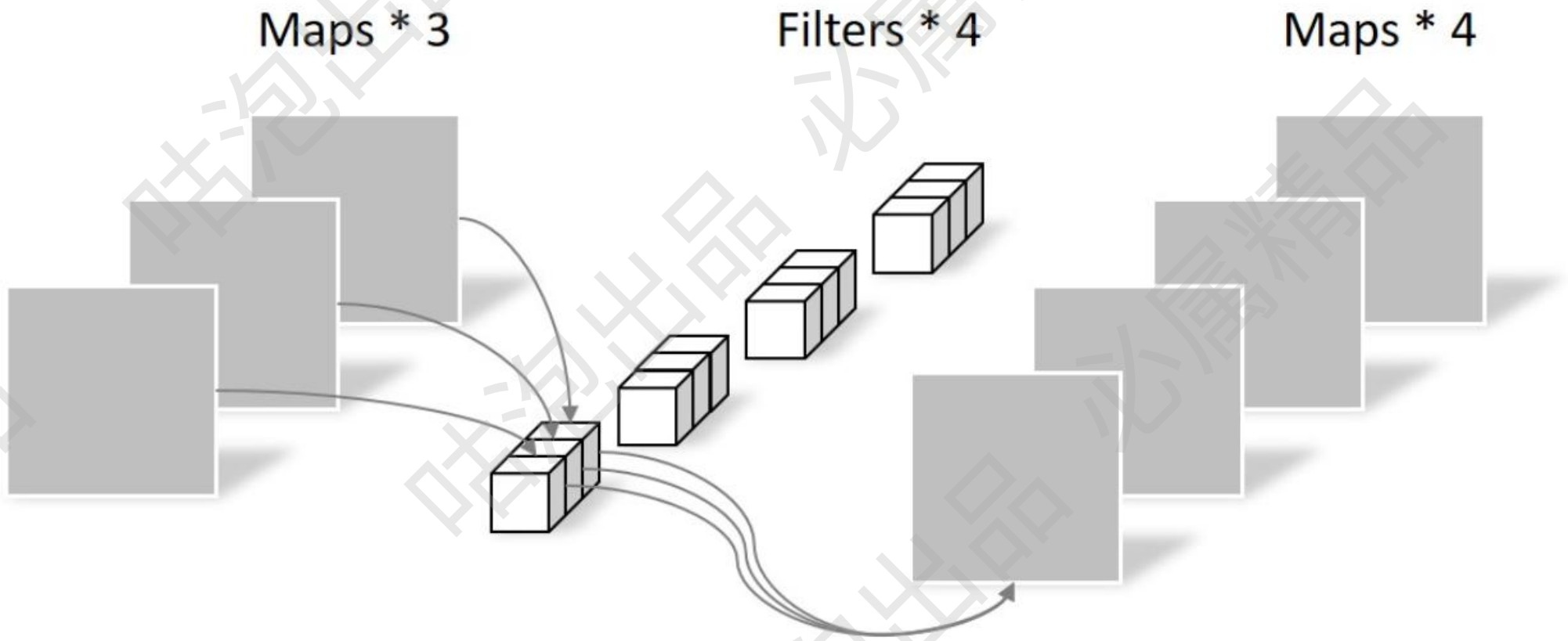
Separation

- Repeat中加入了Depthwise卷积



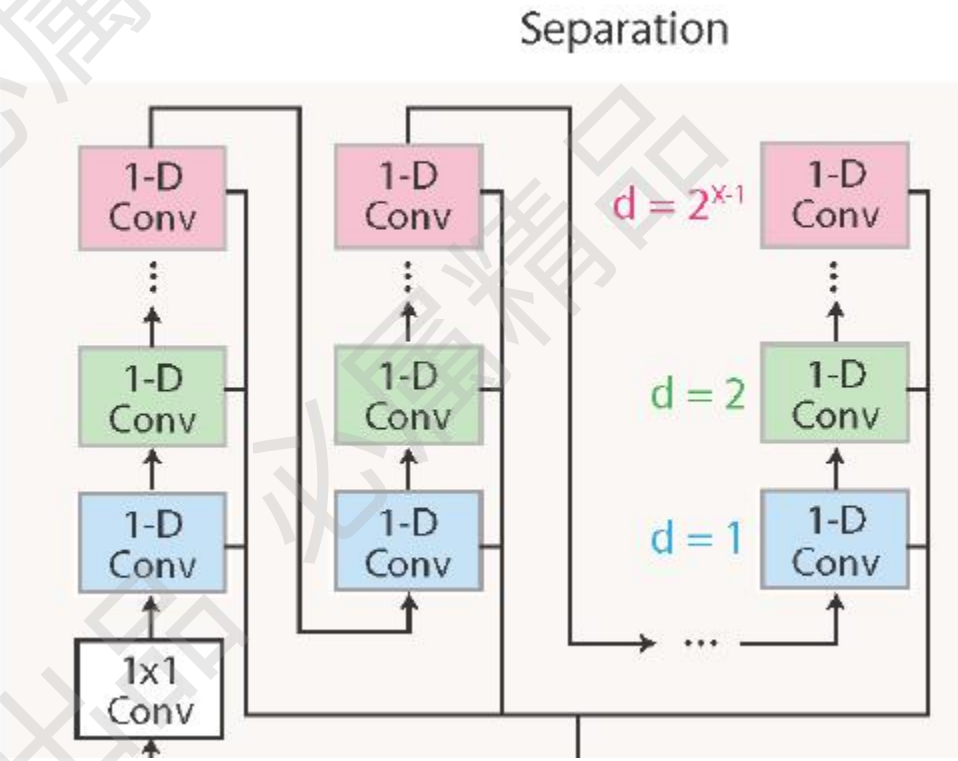
Separation

- Pointwise卷积



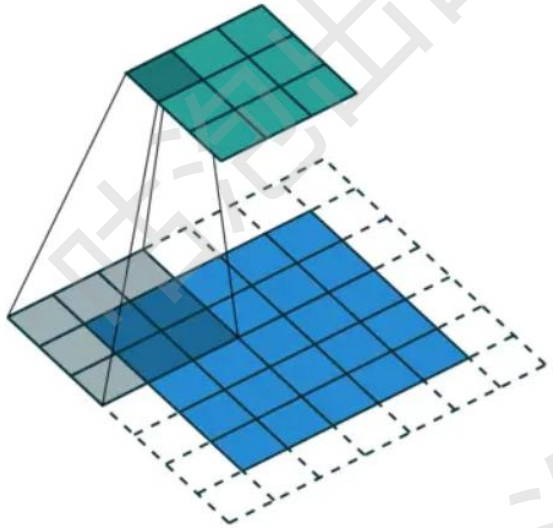
Separation

- 更大的感受野
  - 通过多次重复1-D卷积
  - 并且还是空洞的，这样感受也堆的更快
  - 在堆叠过程中使用DW卷积，省！
  - 很多论文都使用了这种方法

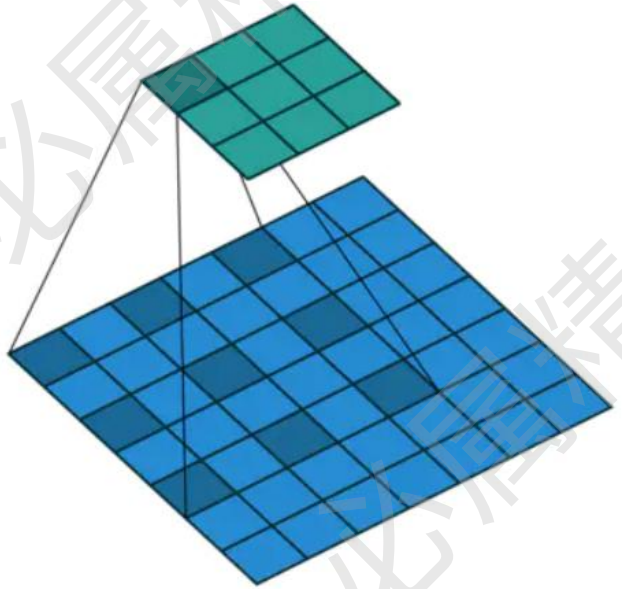


Separation

- 空洞卷积（Dilated/Atrous Convolution）



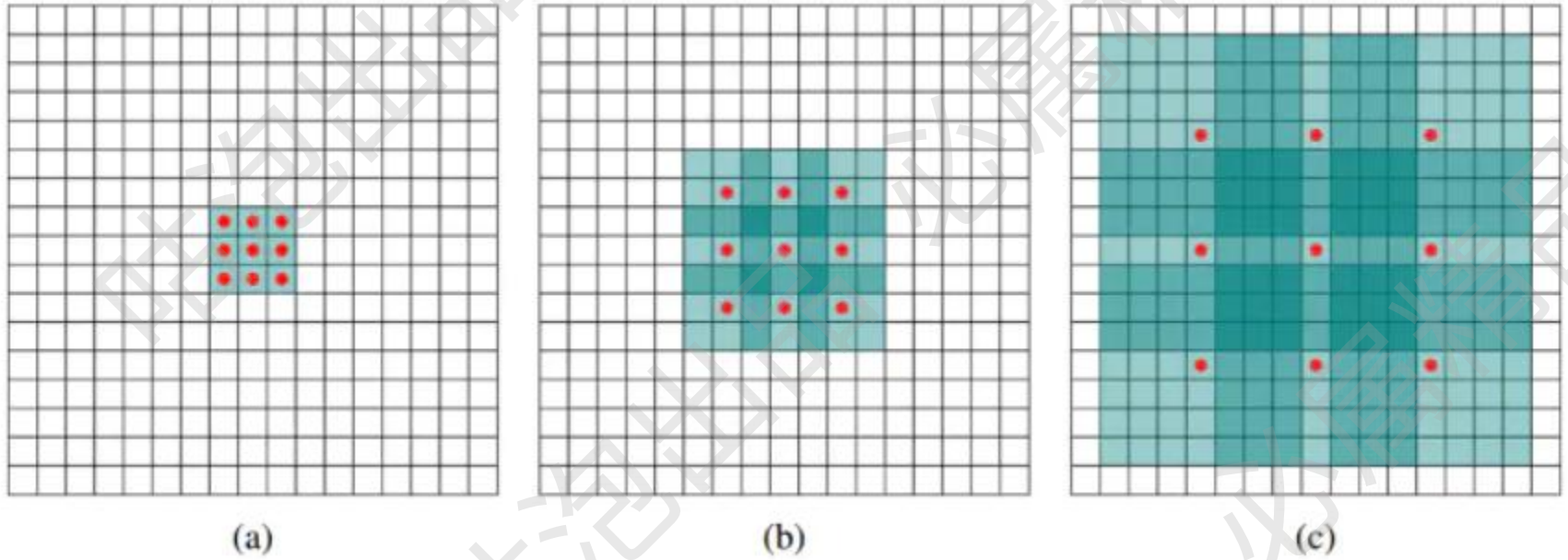
（正常的卷积，按顺序一个个算）



（当d=2时的空洞卷积）

Separation

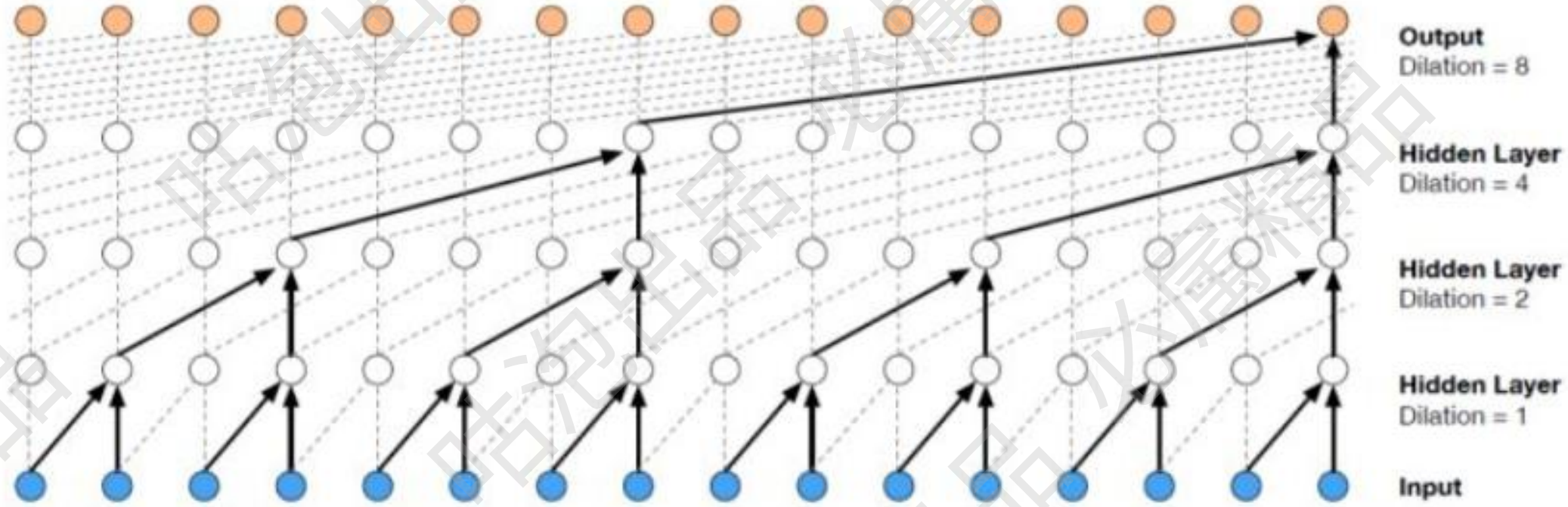
- 当卷积核大小均为3*3时



(正常的感受野)　(2-dilated conv)　(4-dilated conv)

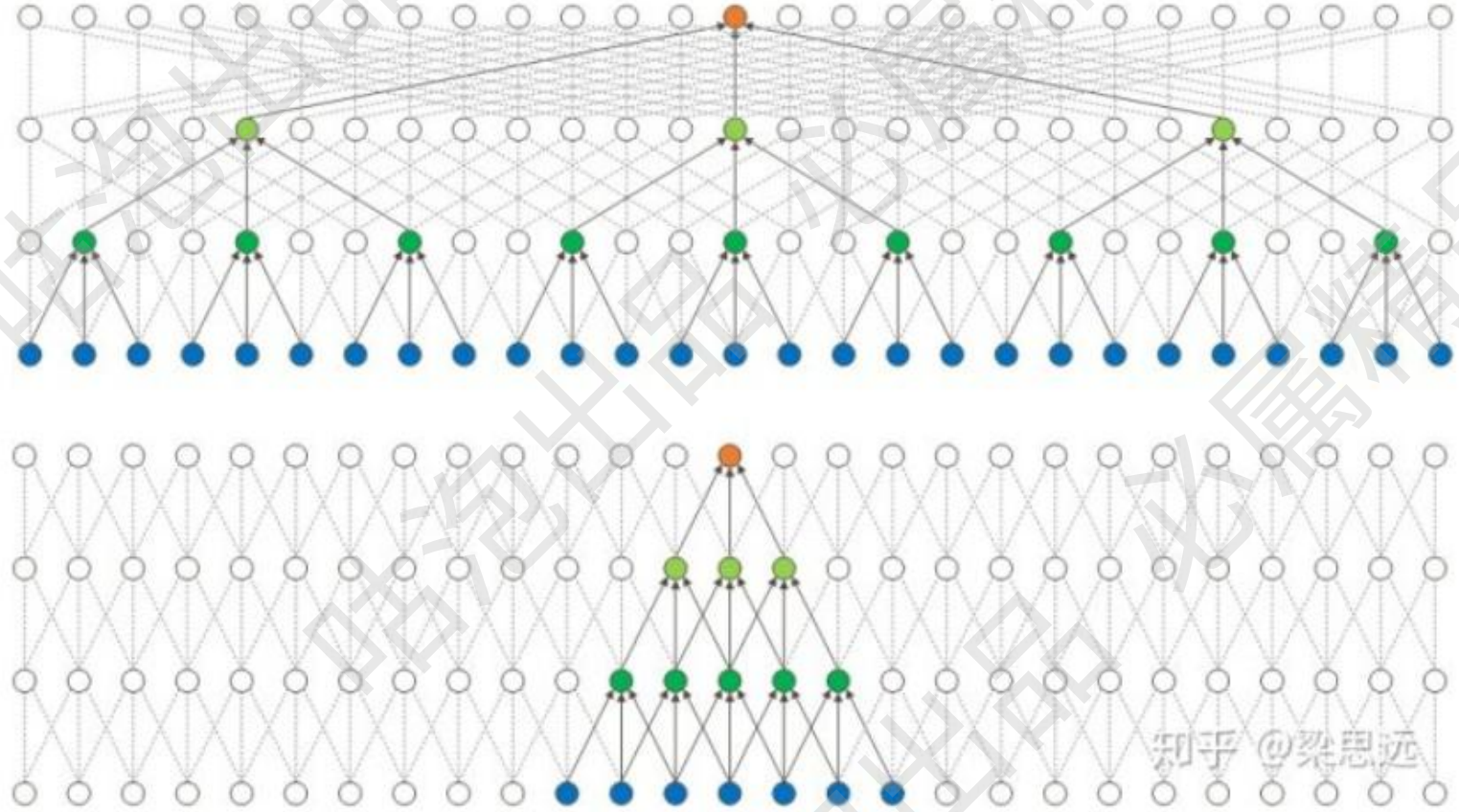
Separation

- 逐渐增大的dilation，我们最终拿到的输出，相当于很大的感受野了！



Separation

- 相当于多次重复之后，我们看到的句子长度在逐渐增大！



Separation

- 评估标准： （scale-invariant source-to-noise ratio）

- 一般用SISNR：

$$\begin{cases} \mathbf{s}_{target} := \dfrac{\langle \hat{\mathbf{s}}, \mathbf{s} \rangle \mathbf{s}}{\|\mathbf{s}\|^2} \\ \mathbf{e}_{noise} := \hat{\mathbf{s}} - \mathbf{s}_{target} \\ \text{SI-SNR} := 10\, log_{10} \dfrac{\|\mathbf{s}_{target}\|^2}{\|\mathbf{e}_{noise}\|^2} \end{cases}$$

$\hat{\mathbf{s}} \in \mathbb{R}^{1\times T}$ and $\mathbf{s} \in \mathbb{R}^{1\times T}$ are the estimated and original clean sources, respectively

（其实就是先把预测结果头像到真实值上得到target,再做个减法就得到noise）

Method	Model size	Causal	SI-SNRi (dB)
Conv-TasNet-gLN	**5.1M**	×	**15.3**
uPIT-LSTM [7]	46.3M	✓	–
LSTM-TasNet [26]	32.0M	✓	**10.8**
Conv-TasNet-cLN	**5.1M**	✓	10.6