Lecture 13:CNN

2018 年 7 月 31 日

1 CNN

1.1 为什么要使用 CNN

RNN 缺陷:

- 在没有上下文的情况下，RNN 无法捕捉 phrase 信息。
- RNN 最终的隐层状态注意力放在了最后的单词上。
- Softmax 也是仅在最后一步使用, 忽视了文本中间部分的信息。

CNN 的朴素思想:
给每一个 n-grams 计算词组向量，无论其是否符合语法规则。

1.2 Convolution

对一维情形:

$$(f \cdot g)[n] = \sum_{h=-H}^{H} f[n-h]g[h]$$

称其为大小为 2M 的滤波器。

1.3 Single Layer CNN

Convolution Filter: $w \in \mathbb{R}^{hk}$.

$$c_i = f(w^T x_{i:i+h-1} + b)$$

Result is a Feature Map

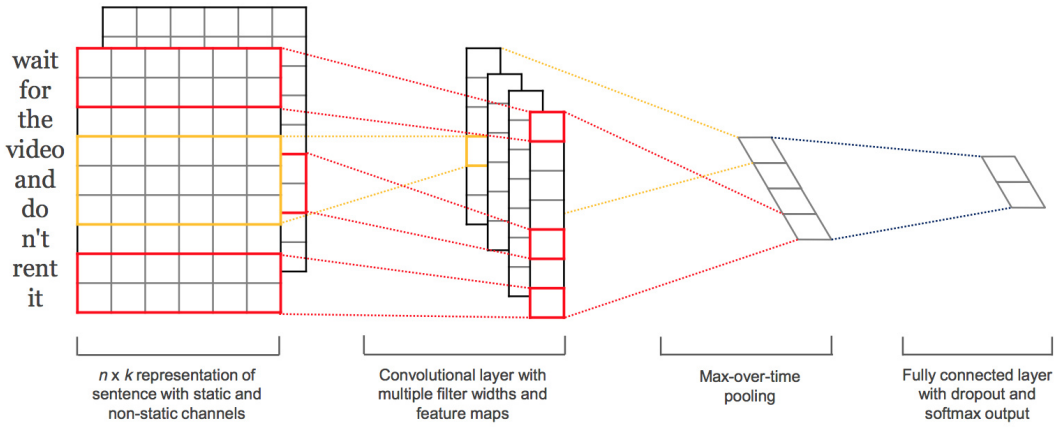
$$c = [c_1, c_2, \cdots, c_{n-h+1}] \in \mathbb{R}^{n-h+1}$$

Pooling:

$$\hat{c} = max\{c\}$$

Prediction: Simple softmax layer.

1.4 Kim 2014

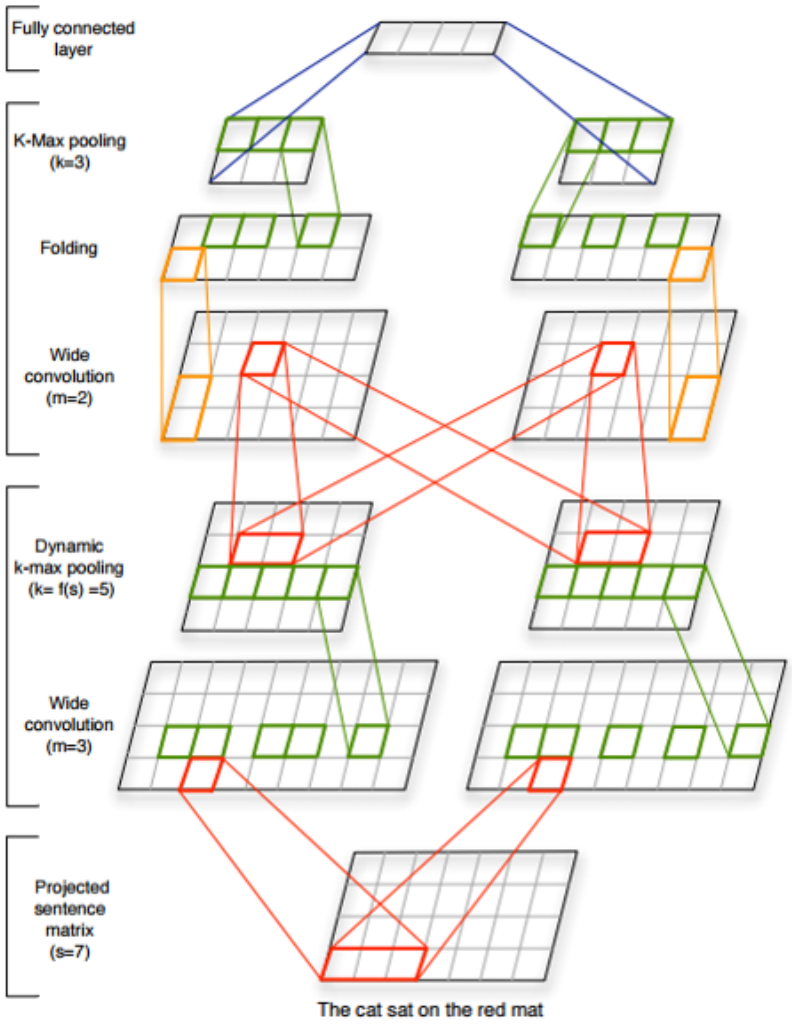


- 双通道词向量，一个 static，另一个随 SGD 更新（因为词向量可能因为分类问题而失去语意的泛化信息）
- 使用多个 Convolution Filter，获得更多的 Features
- m 个卷积核输入 max-pooling 中，得到 $z \in \mathbb{R}^m$
- Dropout：相当于给 GD 引入噪音，使得其搜索更多的状态空间。

$$y = softmax(W^{(S)}(r \circ z) + b)$$

CNN 的核心是提取 local repeating feature，而丢失了序列信息。而 Attention 机制作用在 RNN 上也可以 Focus on 局部信息，且更精准。Attention 是否可以替代 CNN 呢？

1.5 Multi-Layer CNN(Kalchbrenner 2013)

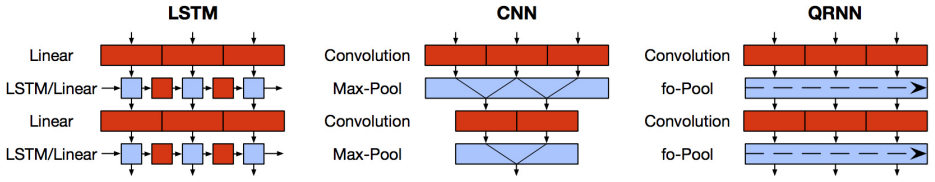


- Wide Convolution: Filter M 将句子中长为 M 所有可能组合都进行卷积。
- K-Max Pooling: 取最大的 k 个值，保留更多信息 + feature 顺序
- Folding：两行 Feature 相加，降维?

CNN 层数越多，池化次数越多，最终学到的特征就越"全局"。
如果想用CNN做局部预测，(如 SQUAD) 该如何做?

1.6 Quasi-RNN(James Bradbury 2017)

- Combines best of both model families



- Convolutions for parallelism across time:

$$\begin{aligned}\mathbf{z}_t &= \tanh(\mathbf{W}_z^1\mathbf{x}_{t-1} + \mathbf{W}_z^2\mathbf{x}_t)\\ \mathbf{f}_t &= \sigma(\mathbf{W}_f^1\mathbf{x}_{t-1} + \mathbf{W}_f^2\mathbf{x}_t)\\ \mathbf{o}_t &= \sigma(\mathbf{W}_o^1\mathbf{x}_{t-1} + \mathbf{W}_o^2\mathbf{x}_t).\end{aligned} \quad \rightarrow \quad \begin{aligned}\mathbf{Z} &= \tanh(\mathbf{W}_z * \mathbf{X})\\ \mathbf{F} &= \sigma(\mathbf{W}_f * \mathbf{X})\\ \mathbf{O} &= \sigma(\mathbf{W}_o * \mathbf{X}),\end{aligned}$$

- Element-wise gated recurrence for parallelism across channels: $\mathbf{h}_t = \mathbf{f}_t \odot \mathbf{h}_{t-1} + (1 - \mathbf{f}_t) \odot \mathbf{z}_t,$

1.6.1 Variables

- Input sequence: $X \in \mathbb{R}^{T \times n}$
- Candidate Vectors: $z_t = tanh(W_z^1 x_{t-1} + W_z^2 x_t)$
- Forget Gate: $f_t = \sigma(W_f^1 x_{t-1} + W_f^2 x_t)$
- Output Gate: $o_t = \sigma(W_o^1 x_{t-1} + W_o^2 x_t)$

Notice: the $*$ denotes a masked convolution along the timestep dimension.

1.6.2 Pooling methods

- $f - pooling$

$$h_t = f_t \odot h_{t-1} + (1 - f_t) \odot z_t$$

- $fo - pooling$

$$c_t = f_t \odot c_{t-1} + (1 - f_t) \odot z_t$$

$$h_t = o_t \odot c_t$$

- $ifo - pooling$

$$h_t = f_t \odot h_{t-1} + i_t \odot z_t$$

$$h_t = o_t \odot c_t$$