Text Summarization

Transformer-based Text Summarization

一、 Transformer 组件

1. Self-attention

Self-attention 和 attention 机制类似，通过向量间的计算，重新分配每个向量的权重，达到关注某些词的效果。在基于 RNN 的 encoder-decoder 模型中，通常会使用 attention 机制对 decoder 中的向量和 encoder 的输出进行计算，由于 RNN 天然的顺序结构，在训练时都是以线性方式处理，无法并行化，所以**训练速度会受限；**其次 RNN 在处理单词时，当前处理单词信息的状态会传递给下一个单词，一个单词的信息会随着距离的增加而衰减，在文本特别长的时候，靠前部分的单词和靠后部分的单词机会没有有效的状态传递，所以**处理长文本的能力弱**。

Transformer 为了决绝 RNN 存在的问题，使用 self-attention，它是对于输入向量进行 attention 计算，可以看作在一个线性投影空间建立输入 X 中不同向量之间的交互关系。在 Transformer 中的 self-attention，使用了三个可训练的参数 W_q, W_k, W_v 对输入进行了线性变换，而多头注意力则是使用 N 组相互独立的 W_q, W_k, W_v 进行 attention 计算，这样可以提取更多的交互信息。

2. Embedding

为了提升效率使用了并行计算，所以句子中的单词失去了原先 RNN 中的先后顺序，因此需要添加 position 标记。在 Transformer 中，使用 Position Encoding 作为附加的 embedding 信息。

3. Residual Network

由于深层网络存在以下两个问题：

1）梯度消失问题

我们发现很深的网络层，由于参数初始化一般更靠近 0，这样在训练的过程中更新浅层网络的参数时，很容易随着网络的深入而导致梯度消失，浅层的参数无法更新。

2）网络退化问题

举个例子，假设已经有了一个最优化的网络结构，是 18 层。当我们设计网络结构的时候，我们并不知道具体多少层次的网络时最优化的网络结构，假设设计了 34 层网络结构。那么多出来的 16 层其实是冗余的，我们希望训练网络的过程中，模型能够自己训练这 16 层为恒等映射，也就是经过这层时的输入与输出完全一样。但是往往模型很难将这 16 层恒等映射的参数学习正确，那么就一定

会不比最优化的 18 层网络结构性能好，这就是随着网络深度增加，模型会产生退化现象。它不是由过拟合产生的，而是由冗余的网络层学习了不是恒等映射的参数造成的。

为了解决上面的问题，**ResNet** 的方法是加上所有跳跃连接，每两层增加一个捷径，构成一个残差块。如图所示，几个残差块连接在一起构成一个残差网络。

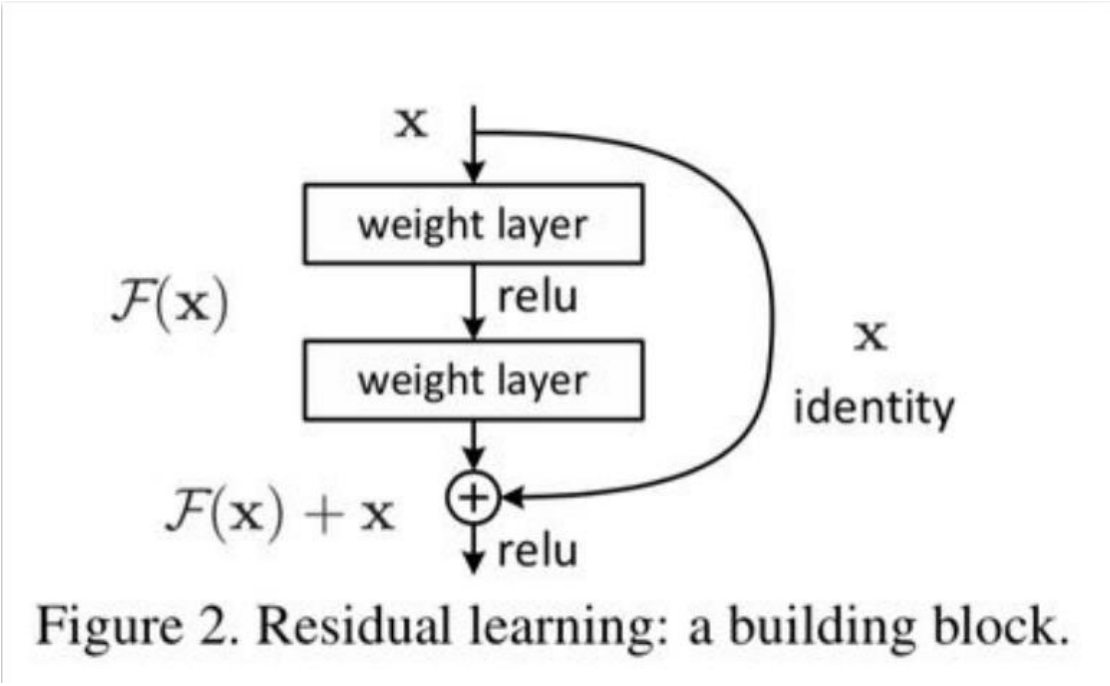


Figure 2. Residual learning: a building block.

1）梯度消失问题

由于最终更新某一个节点的参数时，由于$x_{l+1} = F(x_l, w_l) + x_l$，

则$x_{l+2} = F(x_{l+1},\ w_{l+1}) + x_{l+1} = F(x_{l+1},\ w_{l+1}) + F(x_l, w_l) + x_l$

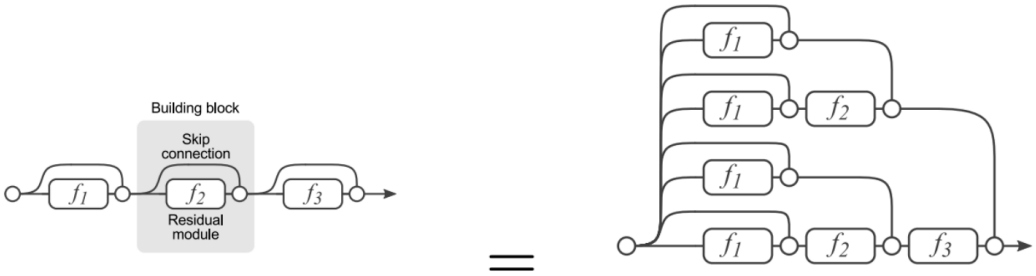
$$x_L = x_l + \sum_{l}^{L-1} F(x_l, w_l)$$

使得不管在哪一层，链式求导后的结果都不会是连乘的形式，而会变成了连加状态，都能保证该节点参数更新不会发生梯度消失或梯度爆炸现象。

2）网络退化问题

假设该层是冗余的，在引入 ResNet 之前，我们想让该层学习到的参数能够满足 h(x)=x，即输入是 x，经过该冗余层后，输出仍然为 x。但是可以看见，要想学习 h(x)=x 恒等映射时的这层参数时比较困难的。ResNet 想到避免去学习该层恒等映射的参数，让 h(x)=F(x)+x;这里的 F(x)我们称作残差项，我们发现，要想让该冗余层能够**恒等映射**，我们只需要学习 F(x)=0。学习 F(x)=0 比学习 h(x)=x 要简单，因为一般每层网络中的参数初始化偏向于 0，这样在相比于更新该网络层的参数来学习 h(x)=x，该冗余层学习 F(x)=0 的更新参数能够更快收敛，网络自行决定了哪些层为冗余层后，通过学习残差 F(x)=0 来让该层网络恒等映射上一层。

3）更好的效果



(a) Conventional 3-block residual network　　(b) Unraveled view of (a)

残差网络就可以被看作是一系列路径集合组装而成的一个集成模型，这表明残差网络展开后的路径具有一定的 独立性和冗余性，使得残差网络表现得像一个集成模型（ensemble），因此，ResNet 的效果也会比普通网络好。

4. Layer Norm

1）为何要 Normalization

因为在模型计算的时候，希望数据差异不要过大，而是独立同分布的，即

（1）去除特征之间的相关性 —> 独立;

（2）使得所有特征具有相同的均值和方差 —> 同分布。

如果层与层之间的分布不一致（Internal Covariate Shif），则会导致

（1）上层参数需要不断适应新的输入数据分布，降低学习速度。

（2）下层输入的变化可能趋向于变大或者变小，导致上层落入饱和 区，使得学习过早停止。

（3）每层的更新都会影响到其它层，因此每层的参数更新策略需要尽 可能的谨慎

2）为何用 layer norm

batch 维度的归一化，也就是对于每个 batch，该层相应的 output 位置归一化所使用的 mean 和 variance 都是一样的。每一个 batch 的句子长度有长有短，如果使用 batch norm 会受到 padding 的值影响。

5. FFN

由于前面的网络结构都是进行线性变换，因此加入一个前馈神经网络增加整个模型的非线性性，并且可以增加模型的记忆能力（来源于论文 Transformer Feed-Forward Layers Are Key-Value Memories）。