

Thinking1: 常用的文本分类方法都有哪些

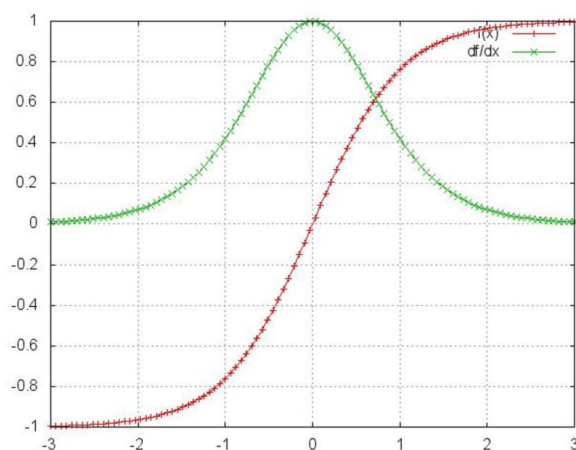
- FastText:原理是把句子中所有的词向量进行平均（某种意义上可以理解为只有一个 avg pooling 特殊 CNN），然后直接接 softmax 层。
- TextCNN: 以捕捉局部相关性，具体到文本分类任务中可以利用 CNN 来提取句子中类似 n-gram 的关键信息。
- TextRNN: 尽管 TextCNN 能够在很多任务里面能有不错的表现，但 CNN 有个最大问题是固定 filter_size 的视野，一方面无法建模更长的序列信息，另一方面 filter_size 的超参调节也很繁琐。CNN 本质是做文本的特征表达工作，而自然语言处理中更常用的是递归神经网络（RNN, Recurrent Neural Network），能够更好的表达上下文信息。具体在文本分类任务中，Bi-directional RNN（实际使用的是双向 LSTM）从某种意义上可以理解为可以捕获变长且双向的 "n-gram" 信息。

Thinking2: RNN 为什么会出现梯度消失

- w 的梯度（T时刻）：

$$\frac{\partial L}{\partial w} = \sum_{t=1}^T \frac{\partial L}{\partial y_T} \frac{\partial y_T}{\partial o_T} \frac{\partial o_T}{\partial S_T} \left(\prod_{k=t+1}^T \frac{\partial S_k}{\partial S_{k-1}} \right) \frac{\partial S_t}{\partial w}$$
$$\prod_{k=t+1}^T \frac{\partial S_k}{\partial S_{k-1}} = \prod_{k=t+1}^T \tanh' W_s$$

Tanh 函数的导数介于[0, 1]之间，如果 W_s 也介于[0, 1]之间，那么当 T 很大的时候，连乘会导致梯度趋近于 0



tanh及其导数图像