慧科 只为赋能人才 | 开课吧 kaikeba.com

文本分类

Jwzheng 慧科Aij讲师

第四节课课程计划

- RNN，LSTM算法回顾
- textRNN模型
- textRNN代码实现（基于keras实现lstm，新闻数据集）

- cnn算法回顾
- textCNN模型
- textCNN代码实现（基于keras实现lstm，新闻数据集）

RNN模型

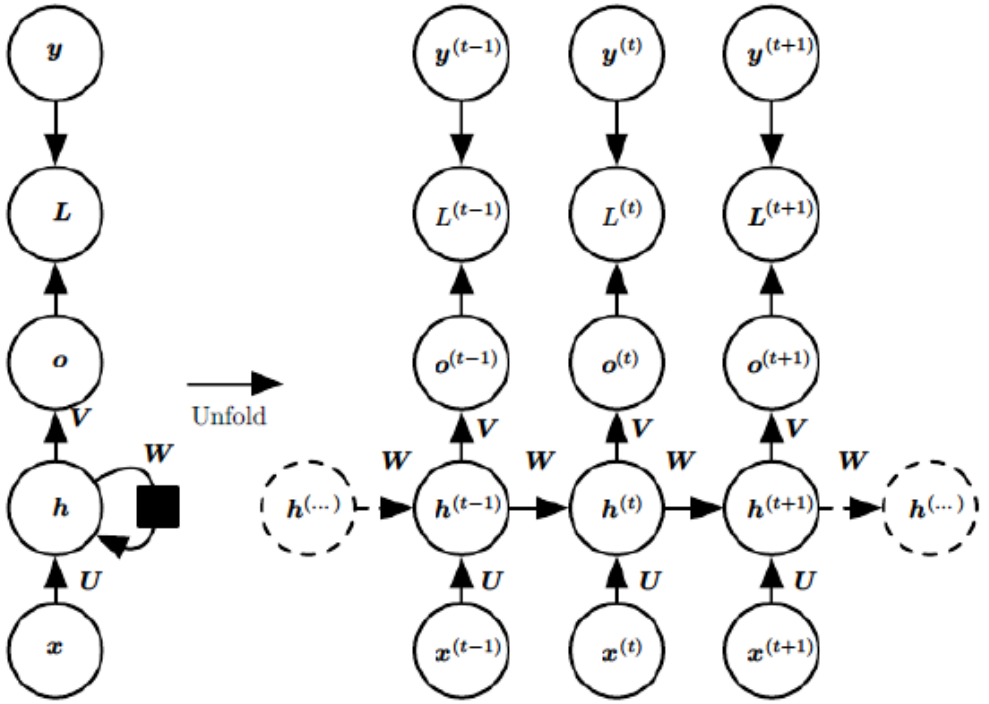
我爸爸是法国人，。。。。。，我喜欢（ ），我们常去法国

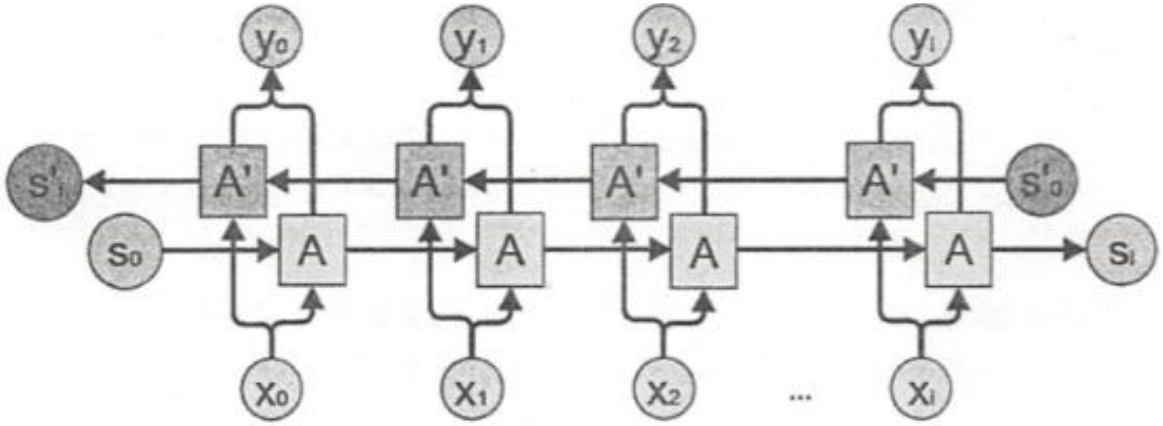
cbow

- RNN网络结构
  - 不同时刻的参数共享

$$h^{(t)} = \phi(Ux^{(t)} + Wh^{(t-1)} + b)$$

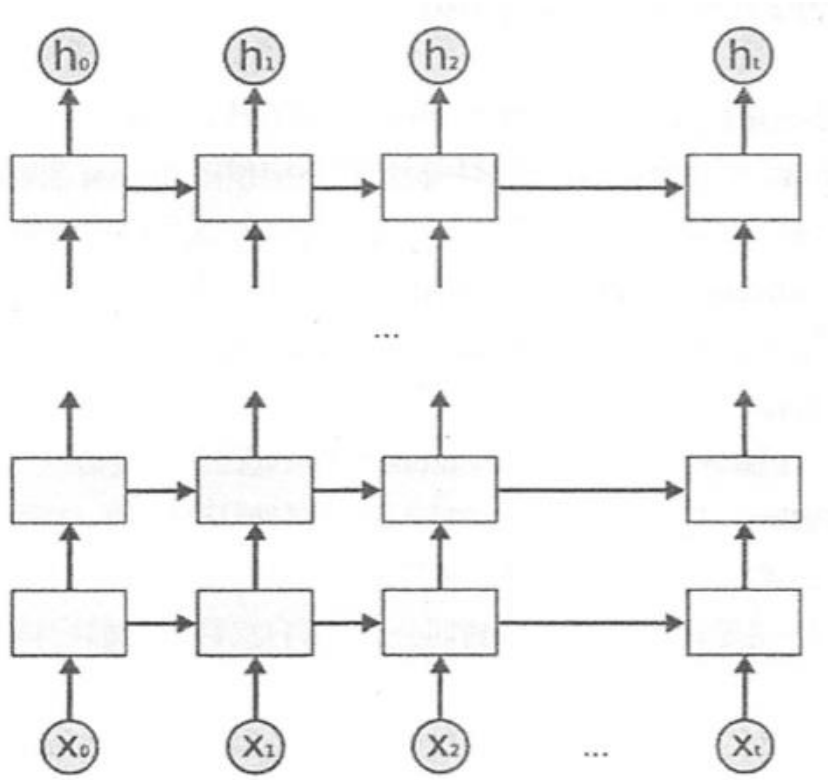
- 双向RNN网络结构
  - 正向和反向不共享参数
  - 双向合并策略
    - 连接 求和





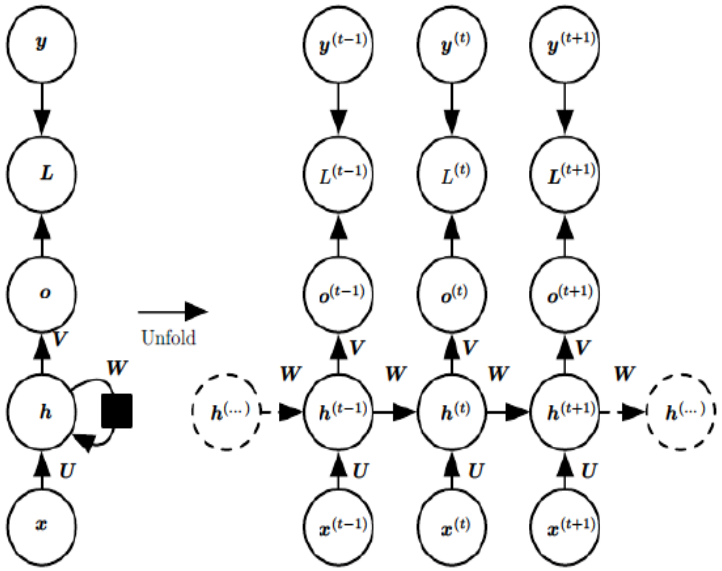
RNN模型

- 深度RNN网络结构
- 深度RNN网络中的dropout
  - 在不同层之间使用

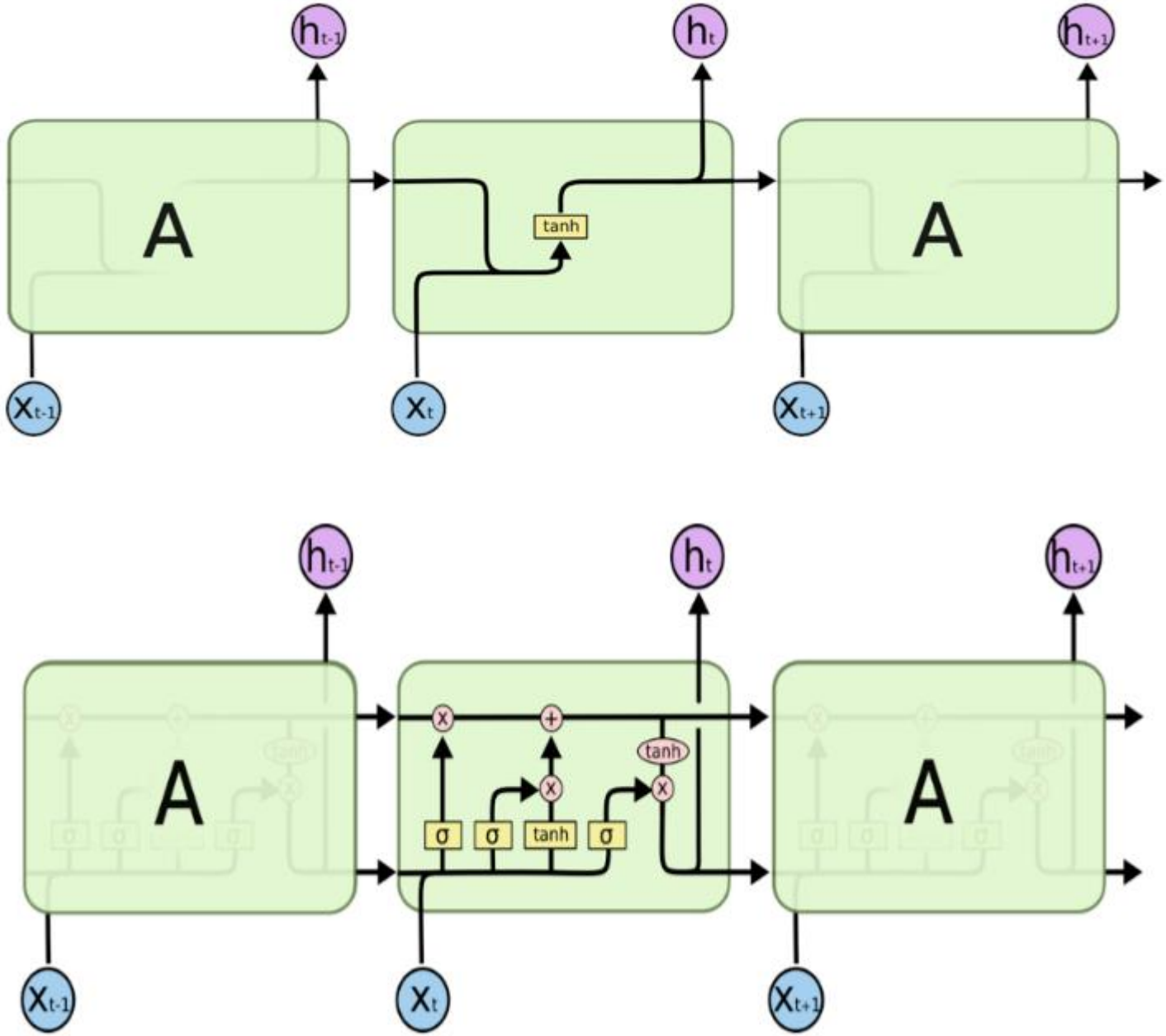


RNN模型

- RNN和LSTM的单元对比



$$h^{(t)} = \phi(Ux^{(t)} + Wh^{(t-1)} + b)$$



LSTM模型

- 遗忘门

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

- 输入门

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$

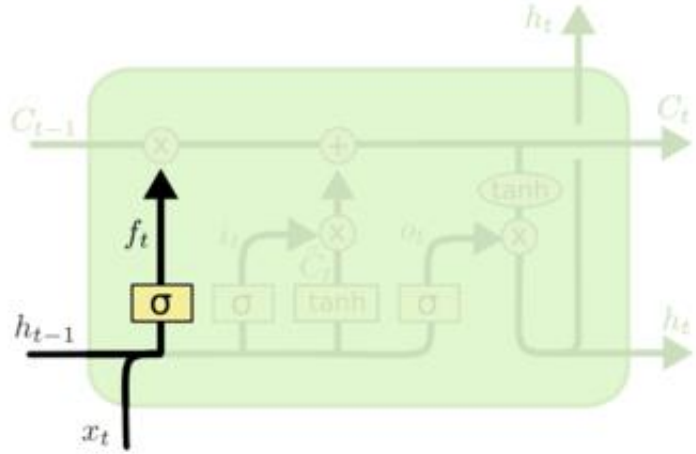
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

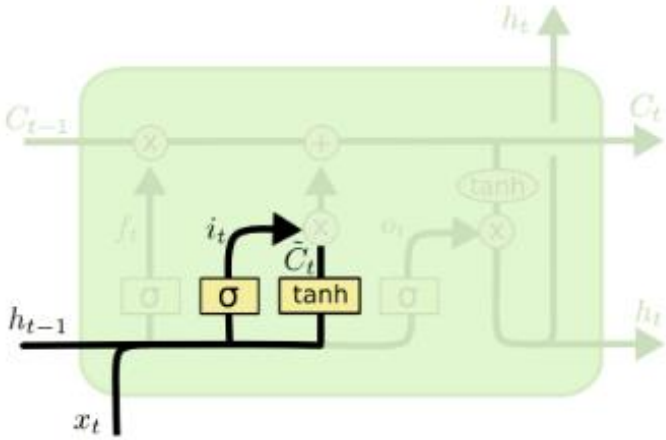
$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

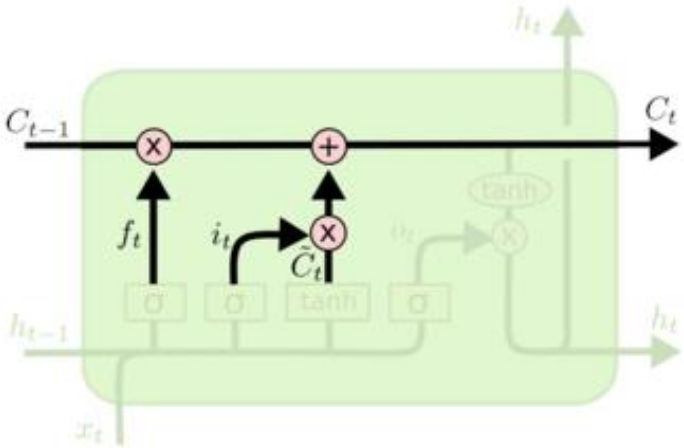
- 输出门

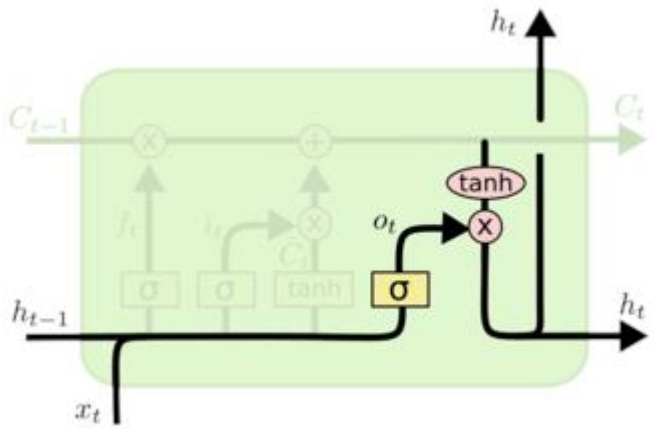
$$o_t = \sigma\left(W_o \; [h_{t-1}, x_t] \; + \; b_o\right)$$

$$h_t = o_t * \tanh\left(C_t\right)$$









TextRNN模型

- RNN(LSTM)用于文本分类
- BiRNN(BiLSTM)用于文本分类
  - 最后一个单词的正向和逆向concat
- embedding层

Layer (type)	Output Shape	Param #
input_1 (InputLayer)	(None, 300)	0
embedding_1 (Embedding)	(None, 300, 200)	4000000
bidirectional_1 (Bidirection	(None, 100)	100400
dropout_1 (Dropout)	(None, 100)	0
dense_1 (Dense)	(None, 10)	1010
dense_2 (Dense)	(None, 10)	110

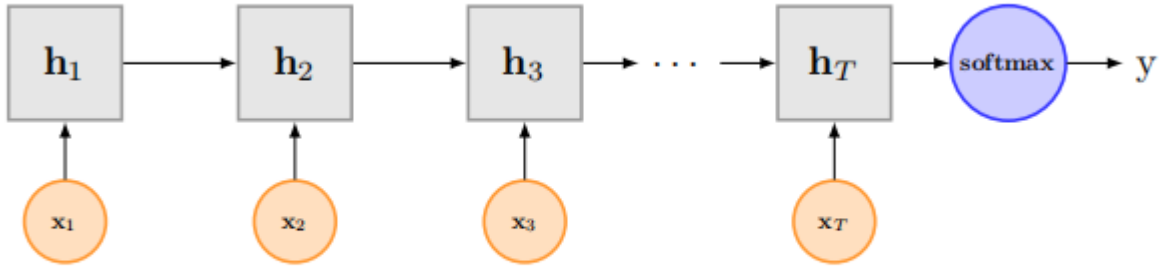
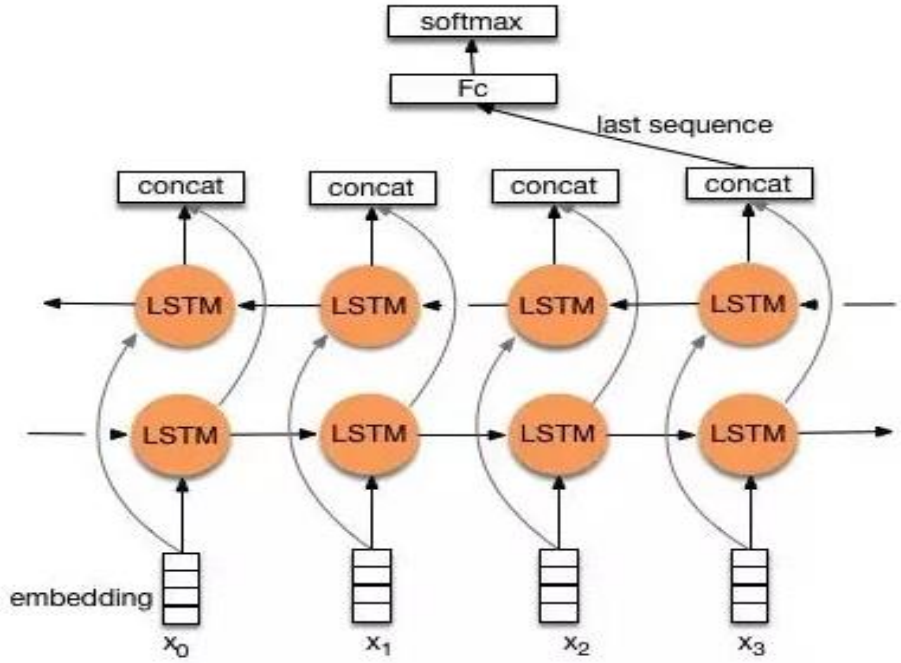
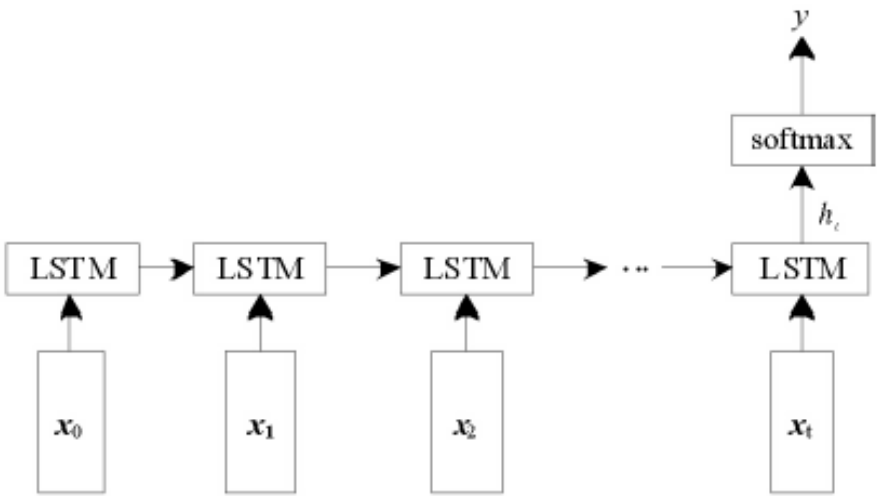


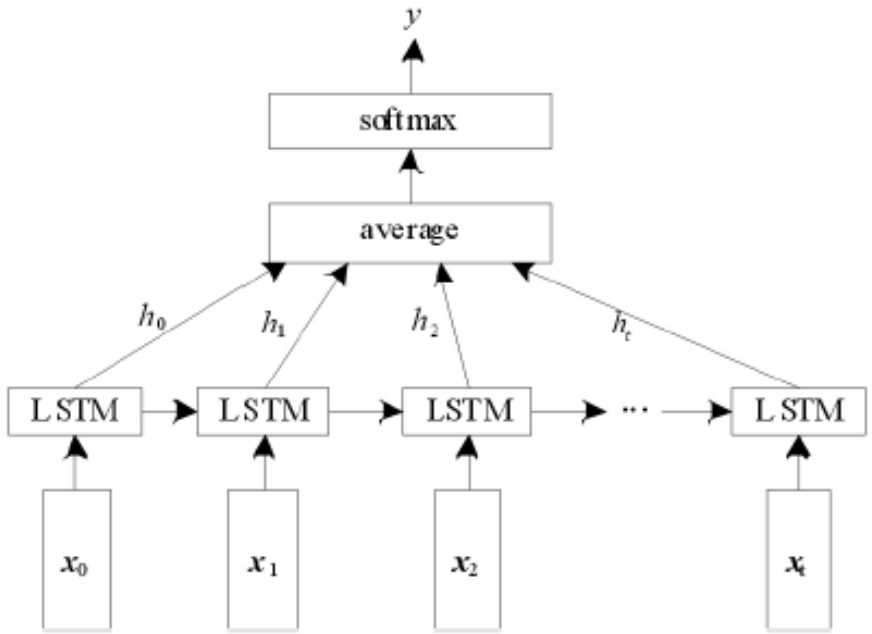
Figure 1: Recurrent Neural Network for Classification



TextRNN模型

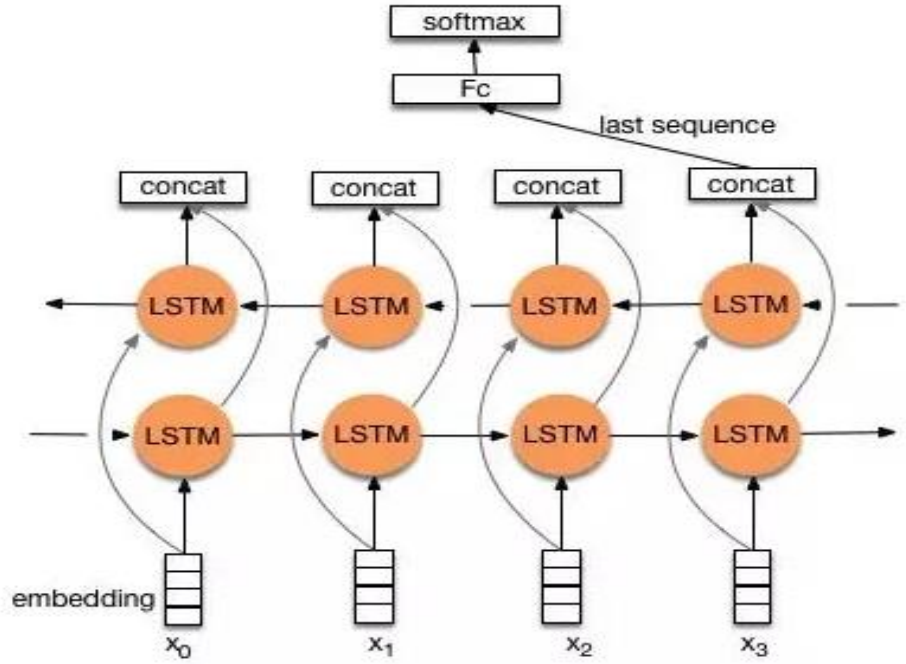
- RNN文本分类

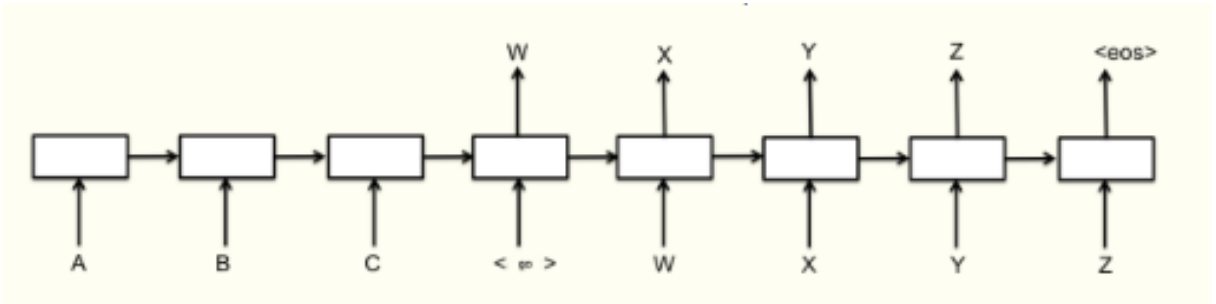




TextRNN模型

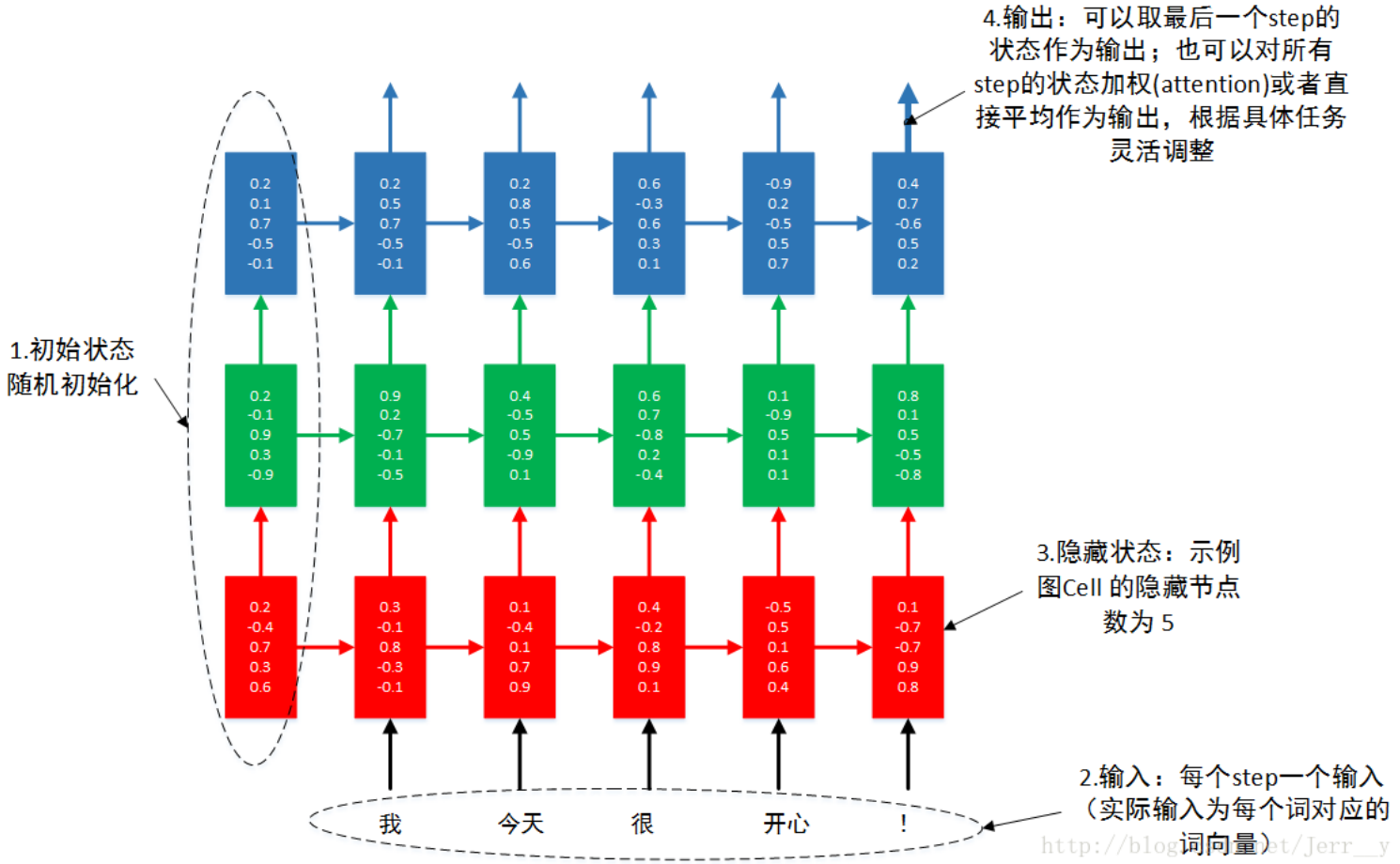
- 为什么使用最后一个向量
  - 整个文本的压缩表示
- 其他方式
  - 使用每个神经元的输出平均后作为文本表示。
- 隐态
  - 计算文档相似度
  - 文档压缩编码
  - 机器翻译
- 端到端的模型
- RNN的文本特征提取





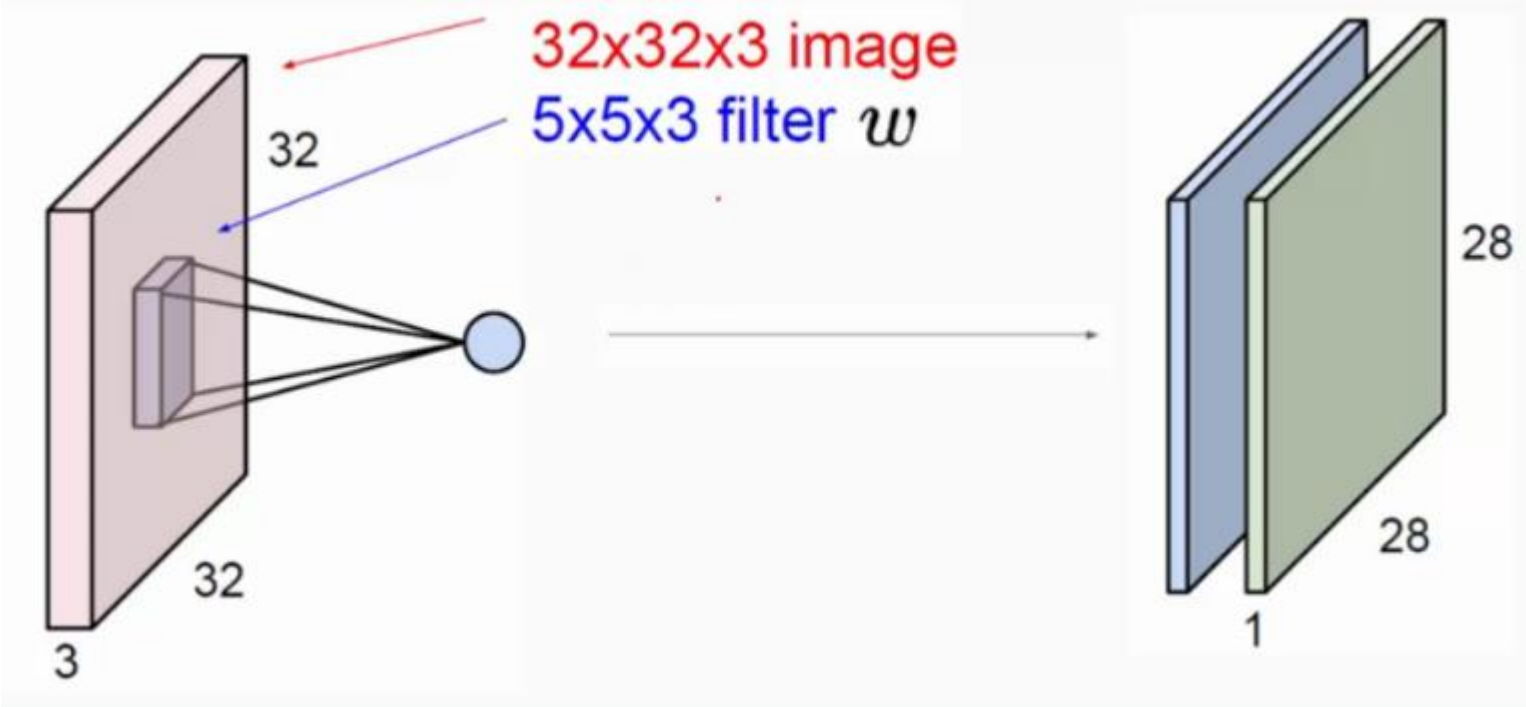
TextRNN模型回顾

- 输入：
  - 一段中文文本（分词）
- 输出：
  - 该文本所属的类别
- 如何表示文本：
  - 词向量表示 word2vec
- 模型：
  - TextRNN,TextBiLSTM



卷积层

- 卷积层是卷积神经网络的核心
- 作用：局部特征提取
- 如何进行卷积？



卷积层

- 灰度图如何进行卷积？

- 6*6图像

步长为1

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

1	−1	−1
−1	1	−1
−1	−1	1

3	−1	−3	−1
−3	1	0	−3
−3	−3	0	1
3	−2	−2	−1

- 6*6的图像

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

1	−1	−1
−1	1	−1
−1	−1	1

卷积核1

−1	1	−1
−1	1	−1
−1	1	−1

卷积核2

- 6*6图像

步长为1

1	0	0	0	0	1
0	1	0	0	1	0
0	0	1	1	0	0
1	0	0	0	1	0
0	1	0	0	1	0
0	0	1	0	1	0

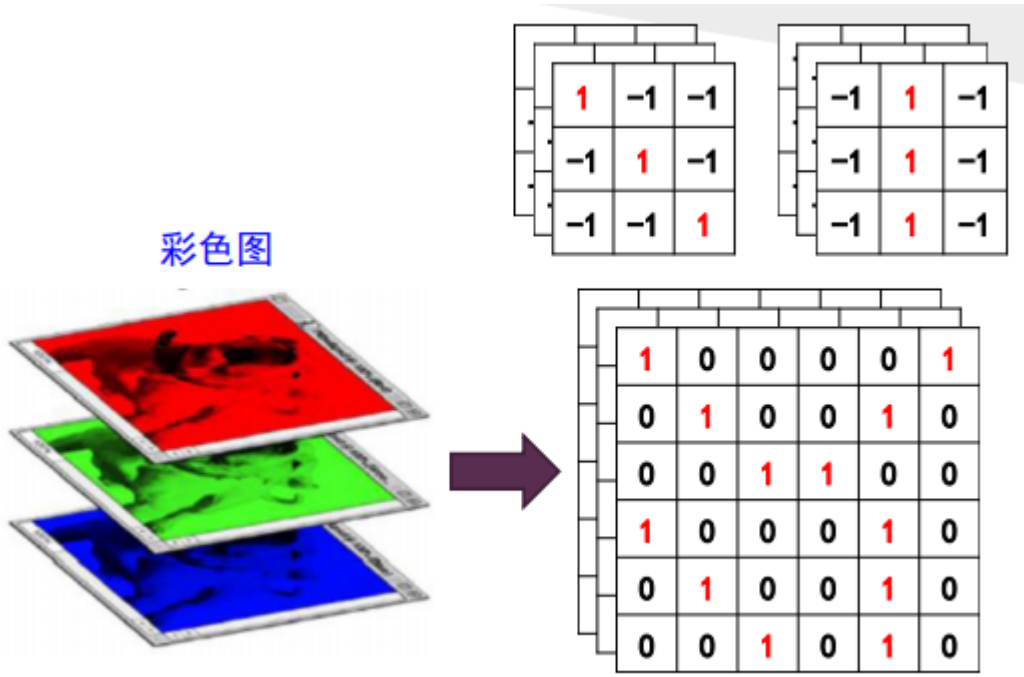
1	−1	−1
−1	1	−1
−1	−1	1

卷积核1

3 −1

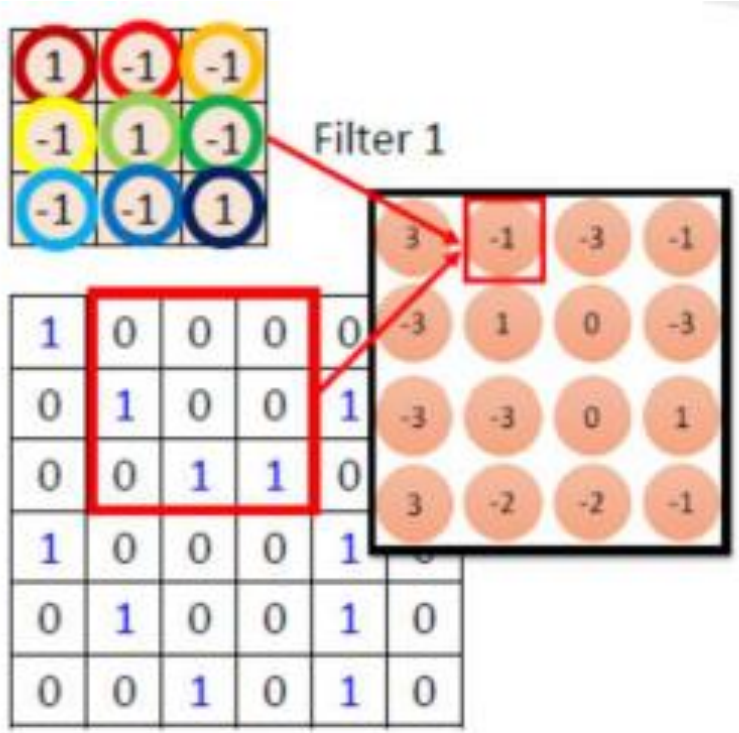
卷积层

- 彩色图如何进行卷积？



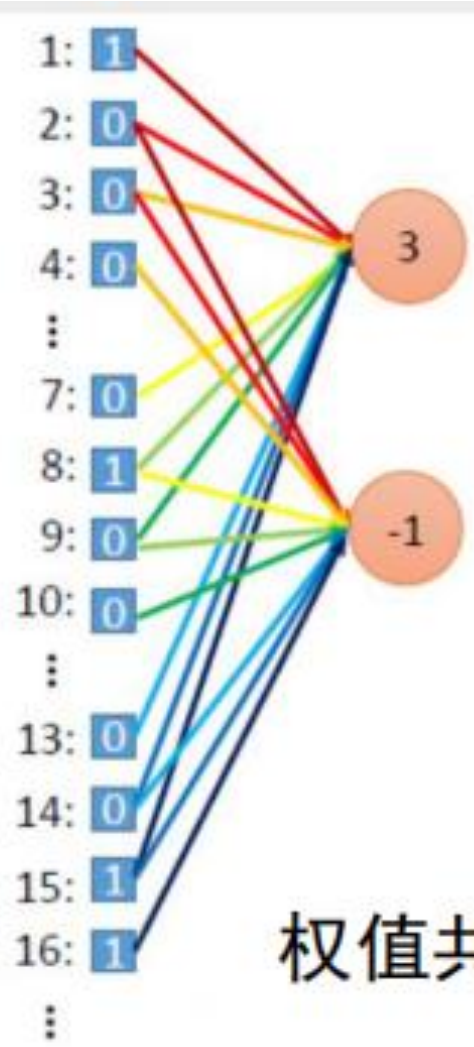
卷积层

- 如何理解卷积？



参数少！

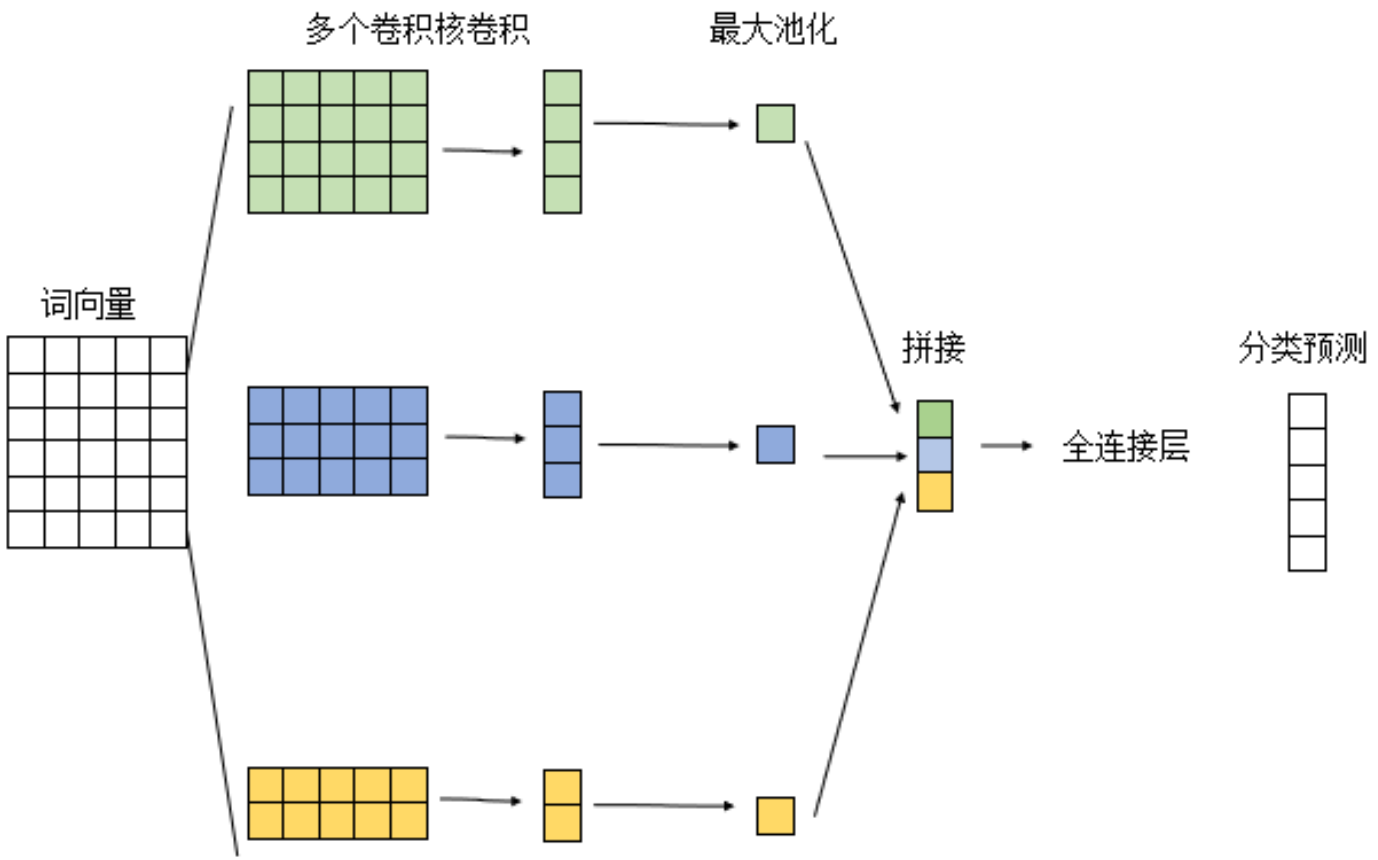
参数其实更少！



权值共享

TextCNN模型

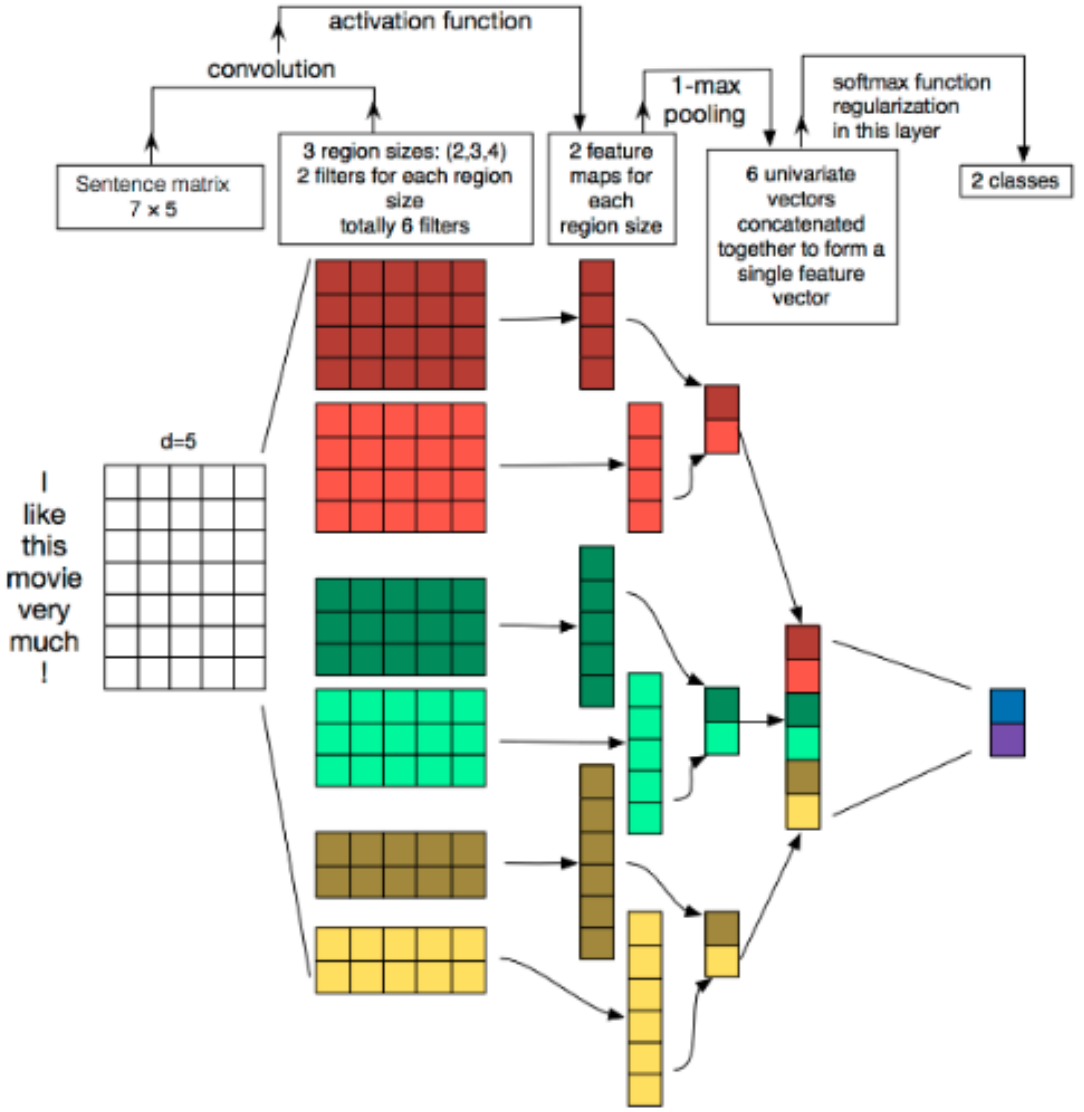
- 输入：
  - 一段中文文本
- 输出：
  - 该文本所属的类别
- 如何表示文本：
  - 词向量表示 word2vec
- 模型：
  - TextCNN



TextCNN模型

(10 100 1000 1001 1002 1003 1004) 7d
7*5d
一句话用矩阵来表示

- 输入：
  - 一段中文文本
- 输出：
  - 该文本所属的类别
- 如何表示文本：
  - 词向量表示 word2vec
- 模型：
  - TextCNN
- 卷积核的维度的意义
  - 长 宽 高



TextCNN模型

- 如何利用多通道
  - 使用多种不同的词向量模型
  - 图像（RGB）

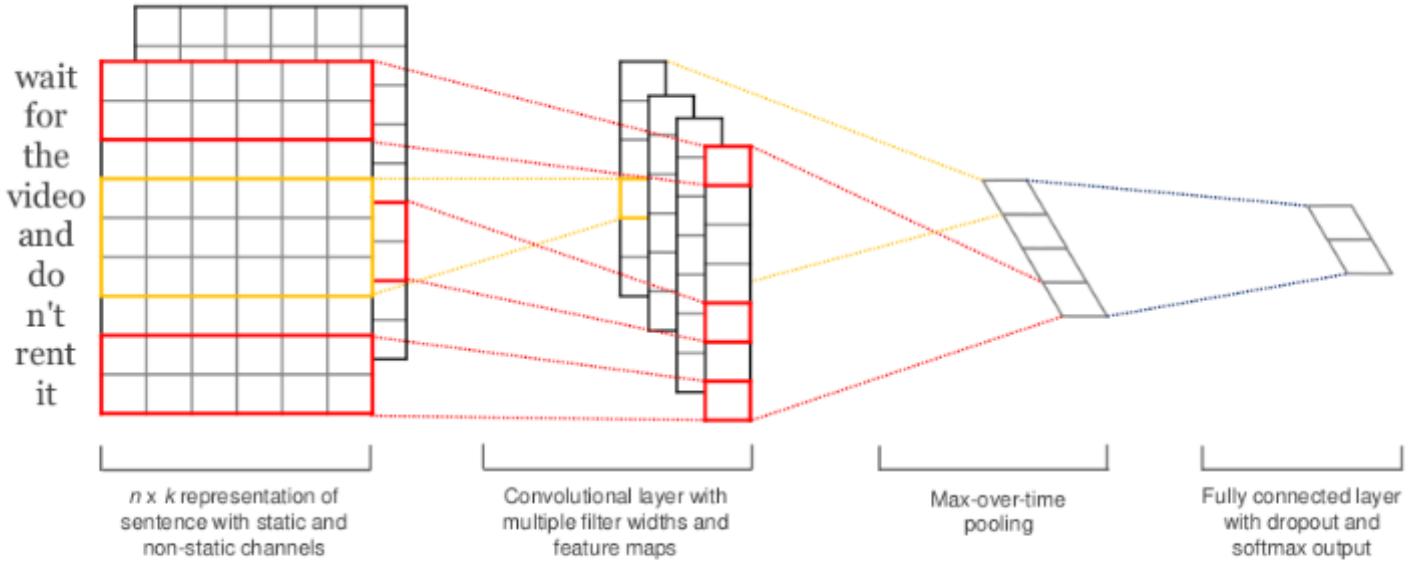


Figure 1: Model architecture with two channels for an example sentence.

第五节课课程计划

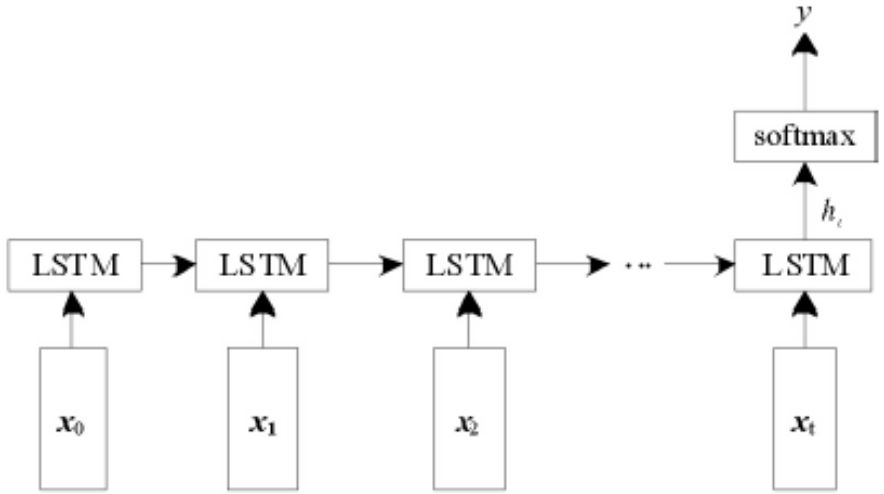
- attention介绍
- lstm+attention文本分类模型
- 实践：lstm+attention文本分类模型

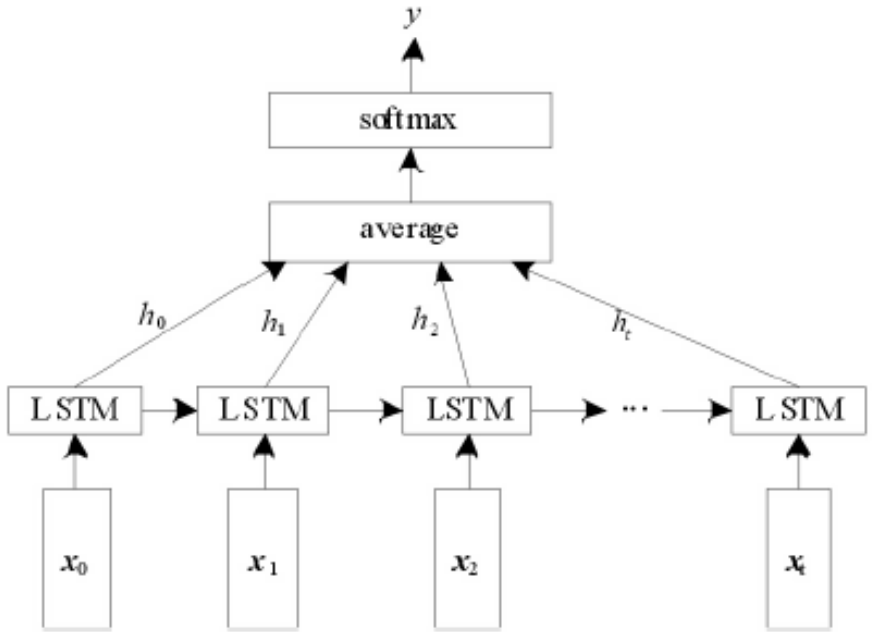
- 分层注意力网络模型介绍
- 实践：基于分层注意力网络实现文本分类

RNN文本分类模型

- RNN文文分类框架
- 缺点
  - 文本表示的欠缺
- 改进
  - 改进的依据
  - 加权重表示文本

$$s = \sum_t \alpha_t h_t$$





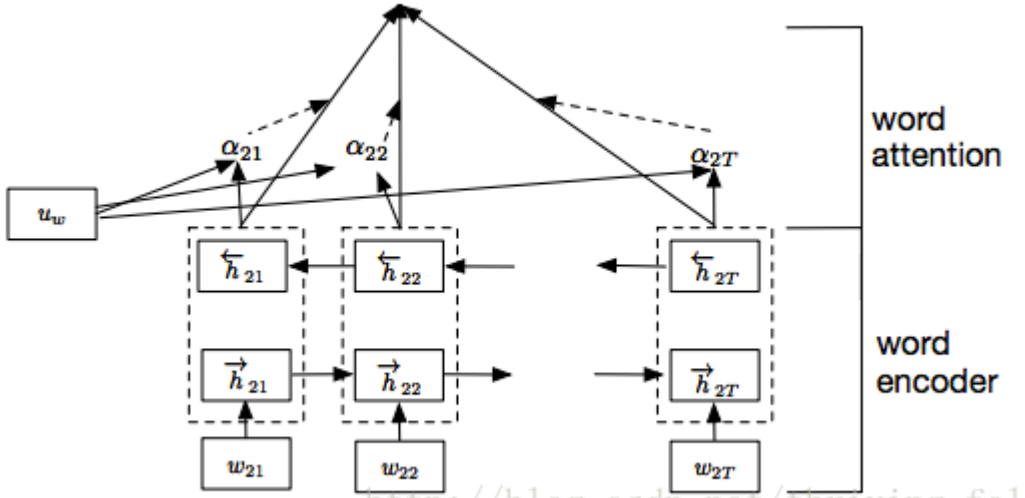
注意力(Attention)

- 如何利用attention进行文本表示

$$u_t = \tanh(W_w h_t + b_w)$$

$$\alpha_t = \frac{\exp(u_t^T u_w)}{\sum_t \exp(u_t^T u_w)}$$

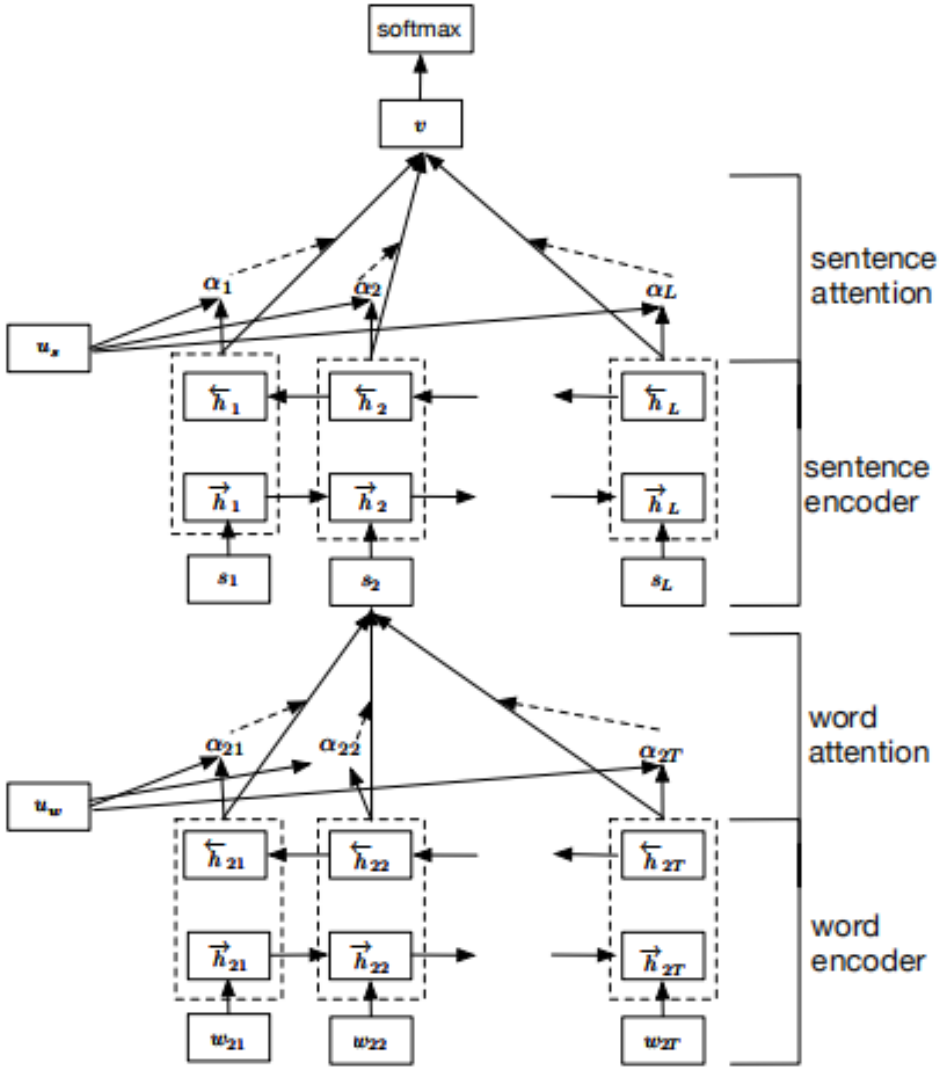
$$s = \sum_t \alpha_t h_t$$



- 公式的理解
  - 参考上面的公式，相当于加了一个attention层，用于计算每一个hi的权重，然后加权各个时刻的隐态，得到文档的表示。

HAN(Hierarchical Attention Networks)

- 网络结构
  - 词序列编码器
  - 基于词级的注意力层
  - 句子编码器
  - 基于句子级的注意力层
- 分层注意力网络用于文本分类
  - 文章由若干个句子组成
  - 句子有若干个单词组成
- GRU作为循环神经网络的每个单元



HAN(Hierarchical Attention Networks)

1）词序列编码器

给定一个句子中的单词 w_{it}，其中 i 表示第 i 个句子，t 表示第 t 个词。通过一个词嵌入矩阵 W_e 将单词转换成向量表示，具体如下所示：

$$x_{it} = W_e w_{it}$$

接下来看看利用双向GRU实现的整个编码流程：

$$\begin{aligned} x_{it} &= W_e w_{it}, t \in [1, T], \\ \overrightarrow{h}_{it} &= \overrightarrow{\mathrm{GRU}}(x_{it}), t \in [1, T], \\ \overleftarrow{h}_{it} &= \overleftarrow{\mathrm{GRU}}(x_{it}), t \in [T, 1]. \end{aligned}$$

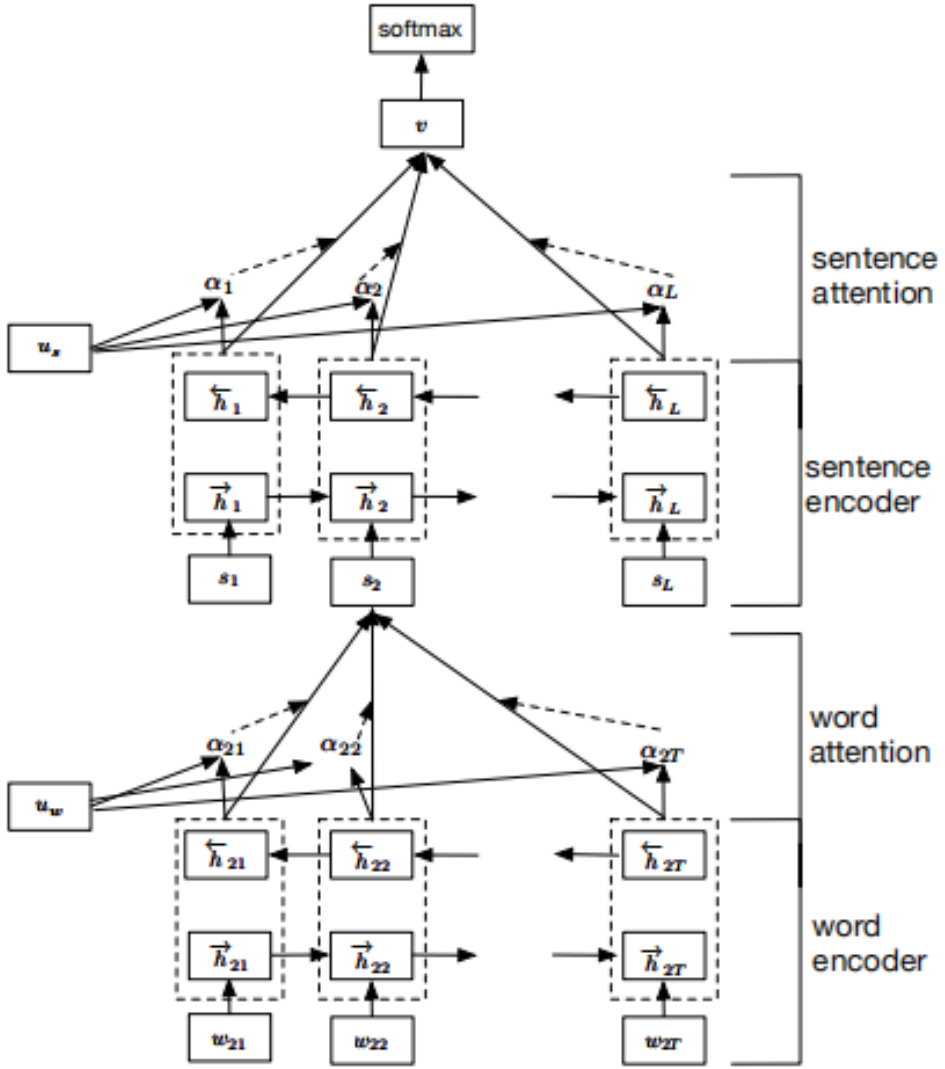
最终的 $h_{it} = [\rightarrow h_{it}, \leftarrow h_{it}]$ 。

2）词级的注意力层

注意力层的具体流程如下：

$$\begin{aligned} u_{it} &= \tanh(W_w h_{it} + b_w) \\ \alpha_{it} &= \frac{\exp(u_{it}^\top u_w)}{\sum_t \exp(u_{it}^\top u_w)} \\ s_i &= \sum_t \alpha_{it} h_{it}. \end{aligned}$$

上面式子中，u_{it} 是 h_{it} 的隐层表示，a_{it} 是经 $softmax$ 函数处理后的归一化权重系数，u_w 是一个随机初始化的向量，之后会作为模型的参数一起被训练，s_i 就是我们得到的第 i 个句子的向量表示。



HAN(Hierarchical Attention Networks)

3）句子编码器

也是基于双向GRU实现编码的，其流程如下，

$$\overrightarrow{h}_i = \overrightarrow{\mathrm{GRU}}(s_i), i \in [1, L],$$
$$\overleftarrow{h}_i = \overleftarrow{\mathrm{GRU}}(s_i), t \in [L, 1].$$

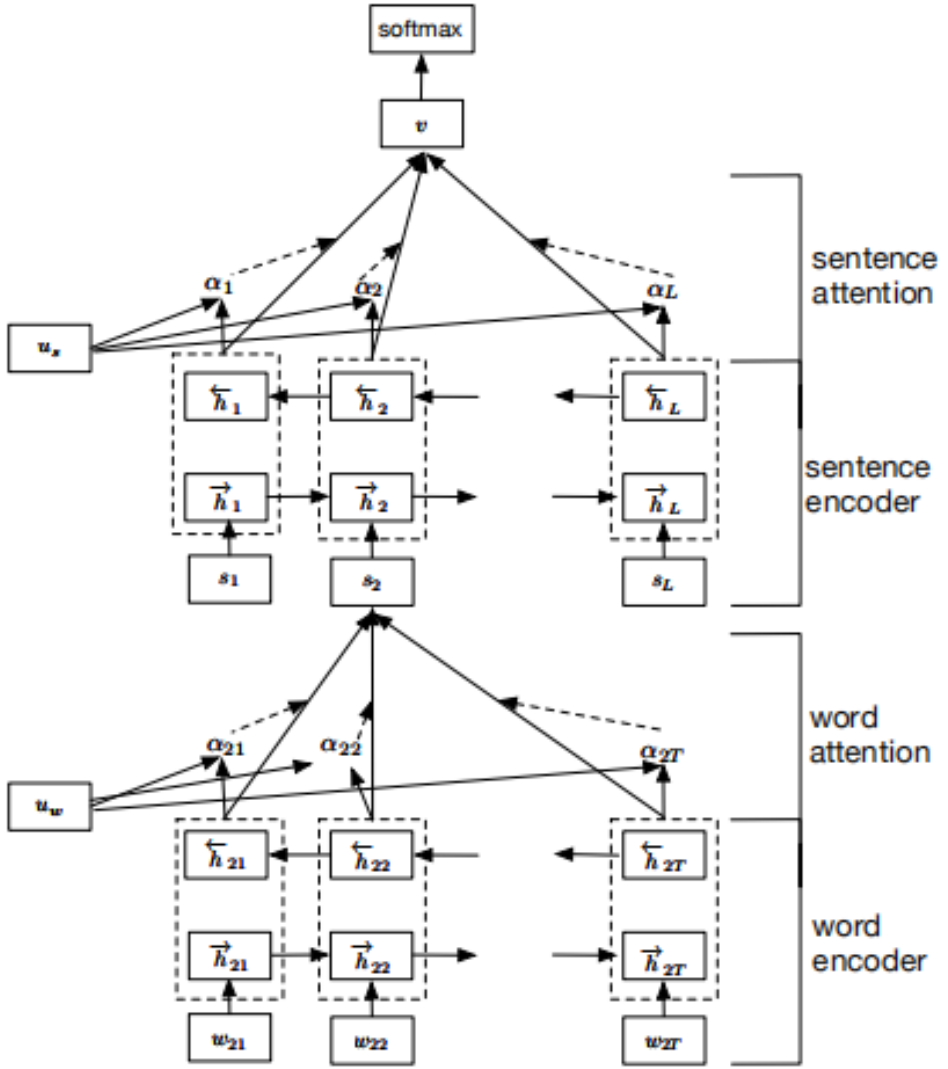
公式和词编码类似，最后的 h_i 也是通过拼接得到的

4）句子级注意力层

注意力层的流程如下，和词级的一致

$$u_i = \tanh(W_s h_i + b_s),$$
$$\alpha_i = \frac{\exp(u_i^\top u_s)}{\sum_i \exp(u_i^\top u_s)},$$
$$v = \sum_i \alpha_i h_i,$$

最后得到的向量 v 就是文档的向量表示，这是文档的高层表示。接下来就可以用可以用这个向量表示作为文



注意力机制的解释

- 源和目标

机器翻译中的Attention

- 源和目标

机器翻译中的Attention

- attention
  - encoder-decoder框架
  - 机器翻译
- 编码

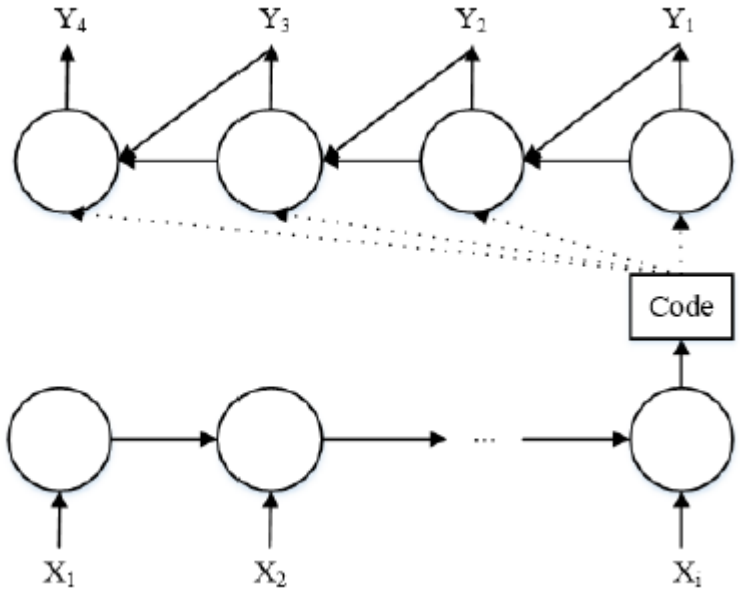
对于一个传统的 Encoder-Decoder 模型，定义输入数据为 $X=(x_1,x_2,x_3,\ldots,x_i)$，输出数据为 $Y=(y_1,y_2,y_3,\ldots,y_j)$，C 为编码后的中间语义，则 Encoder 过程可以表示为：

$$C = E(x_1, x_2, x_3, ..., x_i) \quad (4\text{-}6)$$

- 解码

Decoder 过程可以看作是对中间语义 C 进行解码操作，表示为：

$$y_j = D(C, y_1, y_2, y_3, ..., y_{j-1}) \quad (4\text{-}7)$$

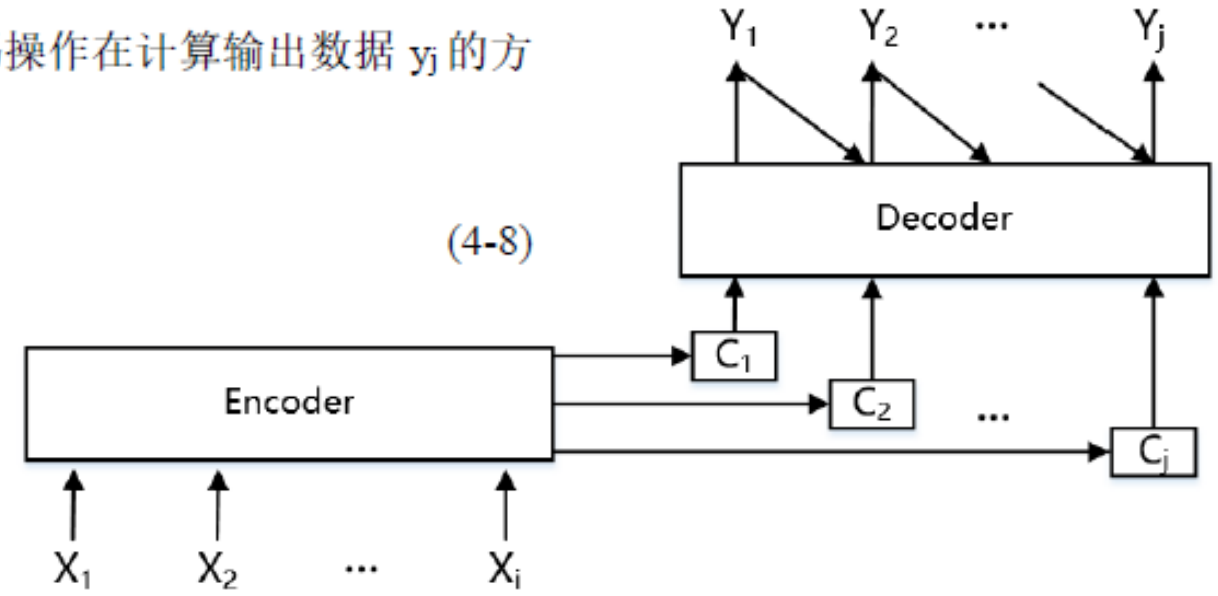


机器翻译中的Attention

- attention

在基于 Attention 机制的 Encoder-Decoder 模型中，解码操作在计算输出数据 y_j 的方式可以表示为：

$$y_j = D(C_j, y_1, y_2, y_3, \cdots, y_{j-1}) \qquad (4\text{-}8)$$



- Cj如何计算

$$C_j = \sum_{i=1}^{T} a_{ij} S(x_i)$$

注意力机制（attention）

- attention的计算

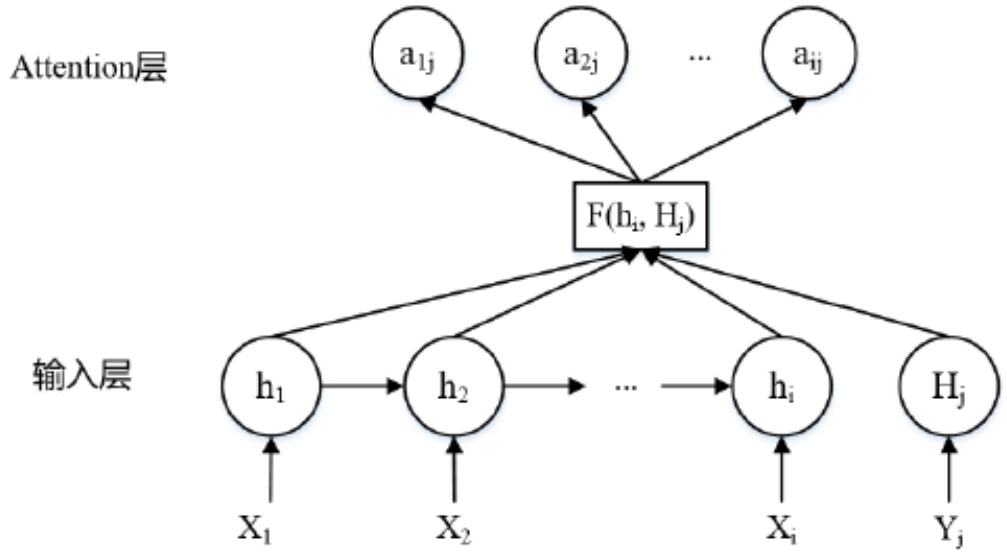


图 4-5 Attention 概率分布的计算示意图

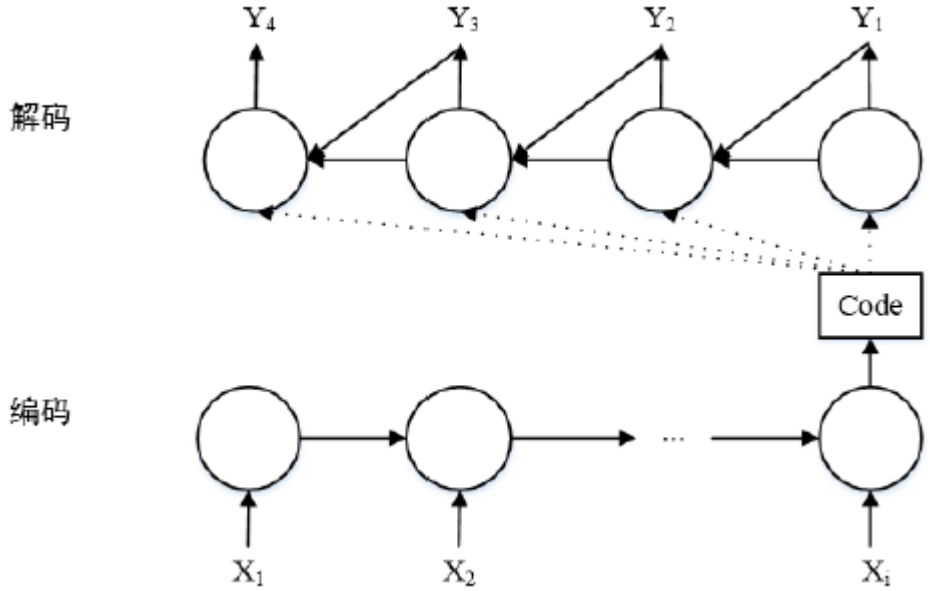
根据图 4-5 中所示，输出值 Y_j 对应输入数据 X_i 的 Attention 概率分布 a_{ij} 的计算方法可以表示为：

$$a_{ij} = F(h_i, H_j) \tag{4-10}$$

其中，h_i 表示输入数据 X_i 在 Encoder 中隐藏层状态，H_j 表示输出数据 Y_j 在 Decoder 中隐藏层状态，F 用于计算两种状态相符合的概率。

注意力机制（attention）

- attention
  - encoder-decoder框架
  - 机器翻译 人机对话
- lstm+attention作文本分类
- bilstm+attention的论文如下：



- Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification

注意力机制（attention）

- Attention原理
  - •计算当前输入序列与输出向量的匹配程度，匹配度高也就是注意力集中点其相对的得分越高。