Deep Learning I

11. 注意力机制

WU Xiaokun 吴晓堃

xkun.wu [at] gmail

2022/05/09

注意力提示

注意力的价值

注意力是**稀缺资源**：价值体现在**机会成本**

- 商业应用：广告、游戏直接宣传
  - 间接模式：会员去广告、付费变强大

注意力的价值

注意力是**稀缺资源**：价值体现在**机会成本**

- 商业应用：广告、游戏直接宣传
  - 间接模式：会员去广告、付费变强大

视觉神经系统：每秒大概接受几十兆信息，**远超大脑处理能力**

- 《史记》：52万字，需要104MB存储
- 大脑的“寄存器”大概只有10个字节

注意力的价值

注意力是**稀缺资源**：价值体现在**机会成本**

- 商业应用：广告、游戏直接宣传
  - 间接模式：会员去广告、付费变强大

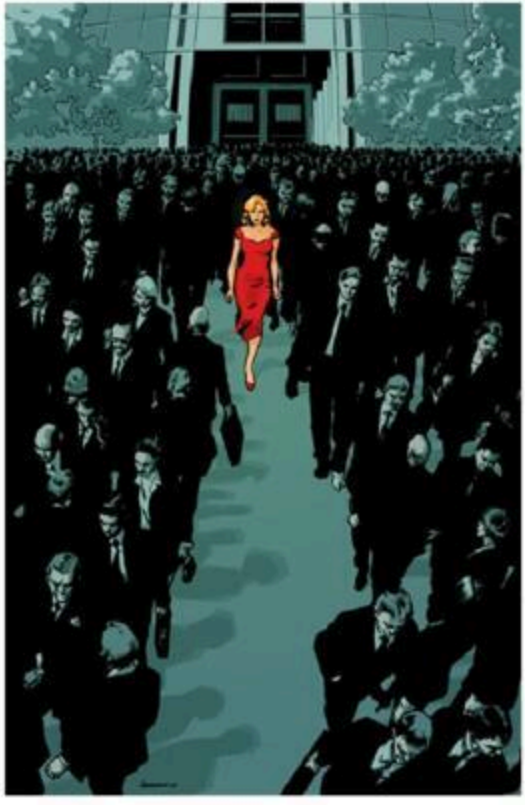
视觉神经系统：每秒大概接受几十兆信息，**远超大脑处理能力**

- 《史记》：52万字，需要104MB存储
- 大脑的“寄存器”大概只有10个字节

- 进化经验：感官输入**并非同等重要**
  - 大脑只关注少量信息，如发现食物、天敌、异性等

注意力：影视

Were you listening to me, Neo? Or were you looking at the woman in the red dress?
– The Matrix

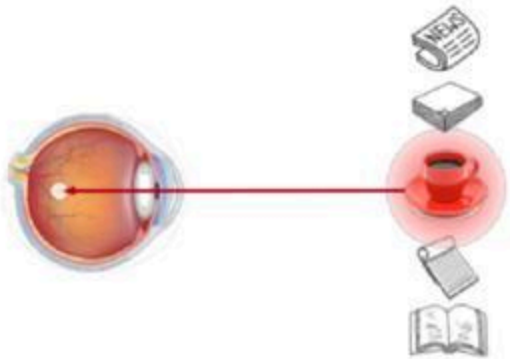


注意力：生物学解释

双组件框架：自主性、非自主性

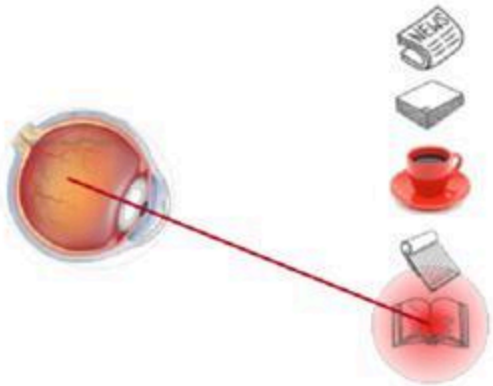
非自主性：基于物体的突出性、易见性

- 下意识地注意，不需要刻意关注



自主性：基于主观意志

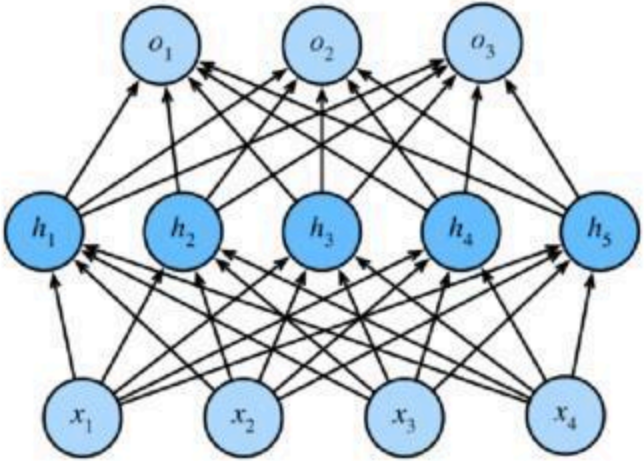
- 认知、意识的控制



非自主性注意力

非自主性：简单全连接，甚至最大池化均可

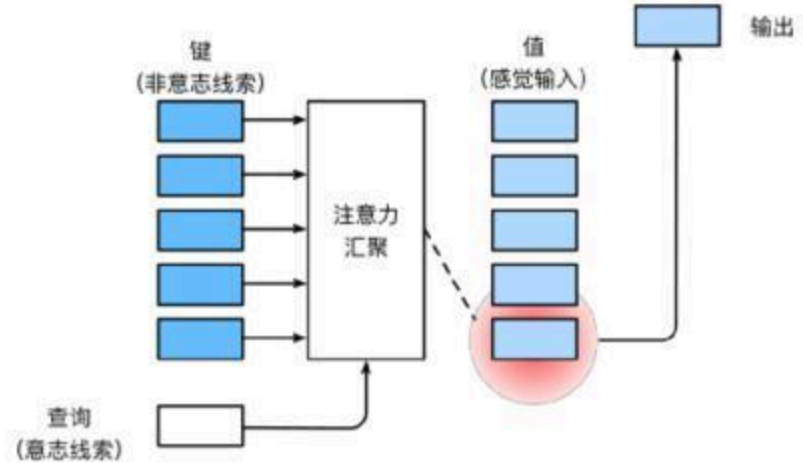
- 选取结果：**自动偏向**高强度感官输入信号



自主性注意力：查询、键、值

主观意志是自主性注意力的关键，可看成**主动查询**

- 键-值对：感官输入的非自主信号



小结：注意力提示

- 注意力是有限、有价的稀缺资源，是人脑资源分配的进化选择
  - 非自主性：基于突出性
  - 自主性：基于主观意志
- 注意力建模
  - 非自主：简单全连接，甚至最大池化
  - 自主：自主提示（查询）

注意力池化

Nadaraya-Watson 核回归

[Nadaraya 1964] [Watson 1964] 根据输入的位置对输出**加权**

$$f(x) = \sum_i \frac{K(x - x_i)}{\sum_j K(x - x_j)} y_i$$

Nadaraya-Watson 核回归

[Nadaraya 1964] [Watson 1964] 根据输入的位置对输出**加权**

$$f(x) = \sum_i \frac{K(x - x_i)}{\sum_j K(x - x_j)} y_i$$

注意力池化：$f(x) = \sum_i \alpha(x, x_i) y_i$

- x：查询；(x_i, y_i)：键-值对

Nadaraya-Watson 核回归

[Nadaraya 1964] [Watson 1964] 根据输入的位置对输出**加权**

$$f(x) = \sum_i \frac{K(x - x_i)}{\sum_j K(x - x_j)} y_i$$

注意力池化：$f(x) = \sum_i \alpha(x, x_i) y_i$

- x：查询；(x_i, y_i)：键-值对
- $\alpha(x, x_i)$：注意力权重，度量x, x_i之间的**相似度**
  - 故f是y_i的加权平均："键"相似，注意力强

Nadaraya-Watson 核回归

[Nadaraya 1964] [Watson 1964] 根据输入的位置对输出**加权**

$$f(x) = \sum_i \frac{K(x - x_i)}{\sum_j K(x - x_j)} y_i$$

注意力池化：$f(x) = \sum_i \alpha(x, x_i) y_i$

- x：查询；(x_i, y_i)：键-值对
- $\alpha(x, x_i)$：注意力权重，度量x, x_i之间的**相似度**
  - 故f是y_i的加权平均："键"相似，注意力强

注意：NW核回归是**非参数模型**；核是超参数，只取决于设计，一般用高斯核

NW 核回归：高斯核

高斯核：高斯函数的简化版本

$$K(u) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{u^2}{2})$$

NW 核回归：高斯核

高斯核：高斯函数的简化版本

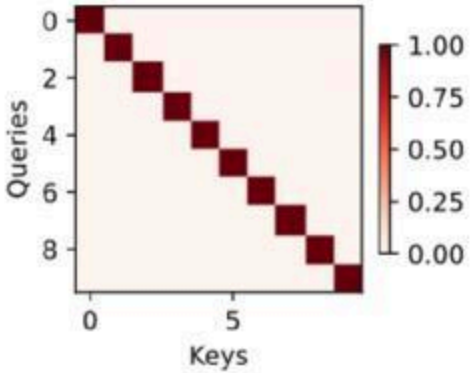
$$K(u) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{u^2}{2})$$

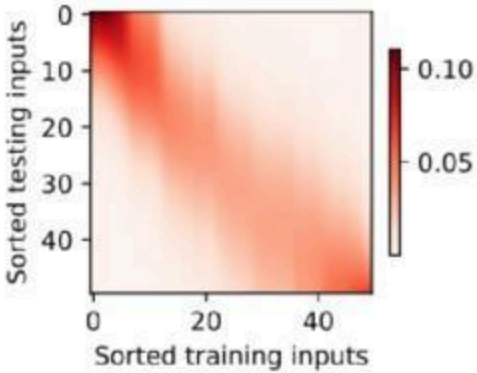
代入注意力池化公式

$$\begin{aligned} f(x) &= \sum_i \alpha(x, x_i) y_i \\ &= \sum_i \frac{\exp(-\frac{1}{2}(x - x_i)^2)}{\sum_j \exp(-\frac{1}{2}(x - x_j)^2)} y_i \\ &= \sum_i \mathrm{softmax}\left(-\frac{1}{2}(x - x_i)^2\right) y_i \end{aligned}$$

注意力权重

测试数据的输入相当于**查询**，训练数据的输入相当于**键**





- 输入经过排序：由观察可知“查询-键”对越接近，注意力权重就越高

带参数注意力池化

非参数 NW 核回归：**一致性**，即不偏向某个“键”x_i

- 只要数据足够多，一定能收敛到最优结果

带参数注意力池化

非参数 NW 核回归：**一致性**，即不偏向某个“键”x_i

- 只要数据足够多，一定能收敛到最优结果

带参数注意力池化：添加**可学习参数**w到度量函数，即区分“键”的重要性

$$
\begin{aligned}
f(x) &= \sum_i \alpha(x, x_i) y_i \\
&= \sum_i \frac{\exp(-\frac{1}{2}((x - x_i)w)^2)}{\sum_j \exp(-\frac{1}{2}((x - x_j)w)^2)} y_i \\
&= \sum_i \mathrm{softmax}\left(-\frac{1}{2}((x - x_i)w)^2\right) y_i
\end{aligned}
$$

实验：NW 核回归

小结：NW 核回归

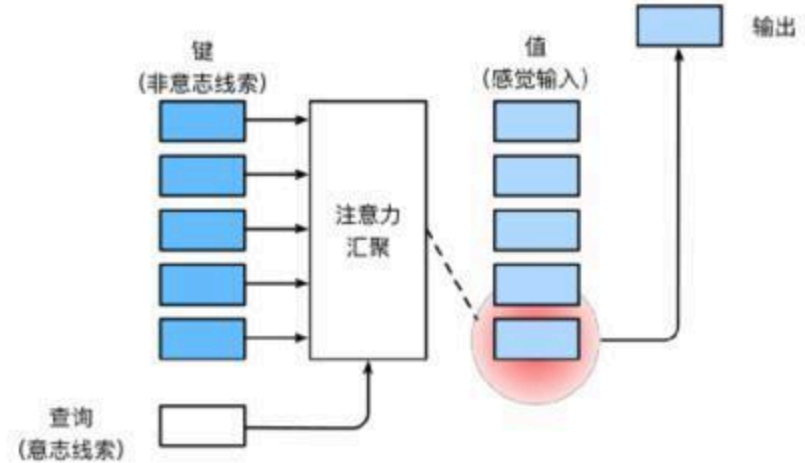
- NW 核回归：具有注意力机制的机器学习范例
- 注意力池化：对训练数据中输出的加权平均
  - 权重：“查询”、“键”之间的相似度
  - 分为非参数型、带参数型

注意力评分函数

回顾：注意力查询、键、值

主观意志是自主性注意力的关键，可看成**主动查询**

- 键-值对：感官输入的非自主信号

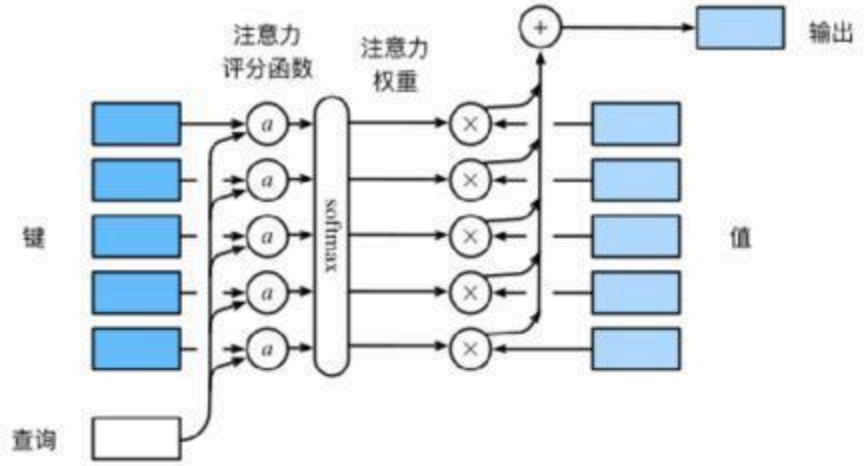


注意力池化：$f(x) = \sum_i \alpha(x, x_i) y_i$

- 可以看成根据“键”的**评分机制**选取“值”

注意力池化：细节

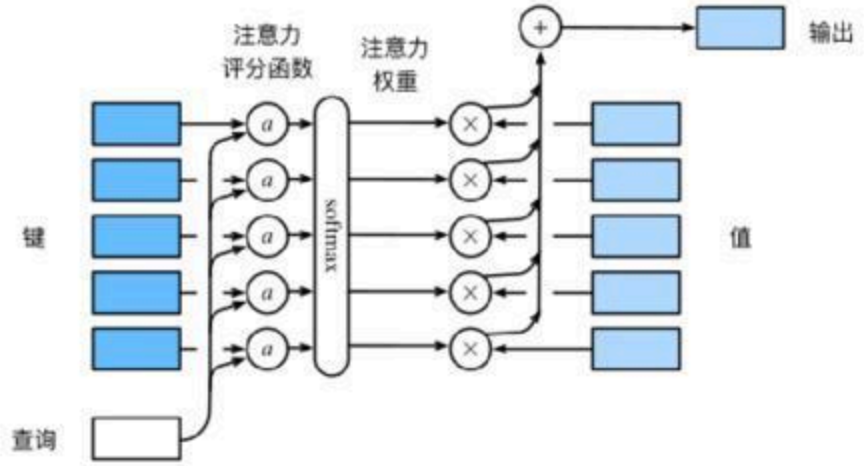
注意力池化： $f(x) = \sum_i \alpha(x, x_i) y_i$



- 核函数$\alpha(x, x_i)$：对“键”x_i的**评分函数**

注意力池化：细节

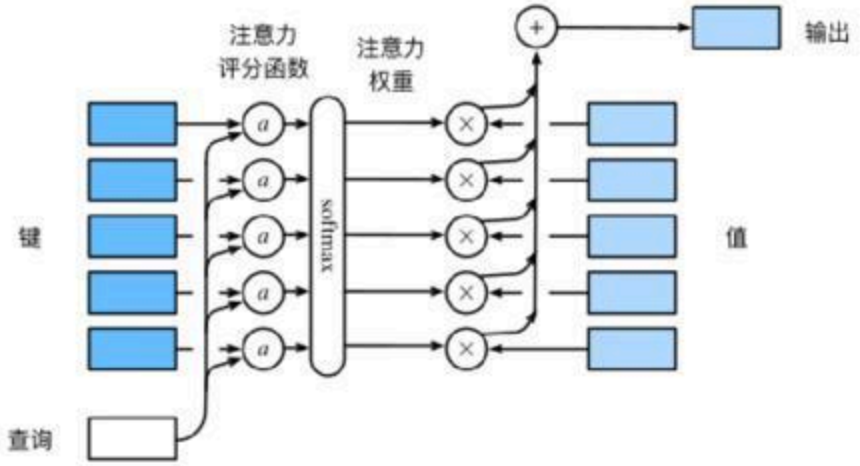
注意力池化： $f(x) = \sum_i \alpha(x, x_i) y_i$



- 核函数$\alpha(x, x_i)$：对“键”x_i的**评分函数**
- `softmax`输出“键”的概率分布，即**注意力权重**

注意力池化：细节

注意力池化：$f(x) = \sum_i \alpha(x, x_i) y_i$



- 核函数$\alpha(x, x_i)$：对“键”x_i的**评分函数**
- `softmax`输出“键”的概率分布，即**注意力权重**
- 输出：基于注意力权重的“值”加权和

注意力池化：计算

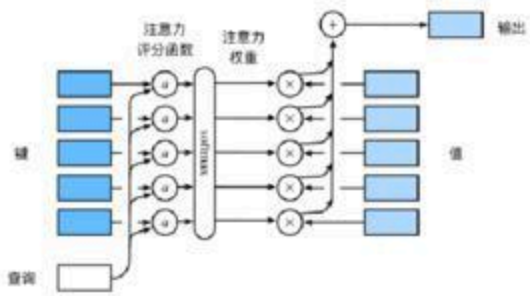
注意力池化：$f(x) = \sum_i \alpha(x, x_i) y_i$

查询$\mathbf{q}$和m个“键－值”对$(\mathbf{k}_m, \mathbf{v}_m)$

$$f(\mathbf{q}, (\mathbf{k}_1, \mathbf{v}_1), .., (\mathbf{k}_m, \mathbf{v}_m)) = \sum_i \alpha(\mathbf{q}, \mathbf{k}_i)\mathbf{v}_i$$

评分函数α

$$\alpha(\mathbf{q}, \mathbf{k}_i) = \mathrm{softmax}(a(\mathbf{q}, \mathbf{k}_i)) = \frac{\exp(a(\mathbf{q}, \mathbf{k}_i))}{\sum_j \exp(a(\mathbf{q}, \mathbf{k}_j}$$



注意力池化：计算

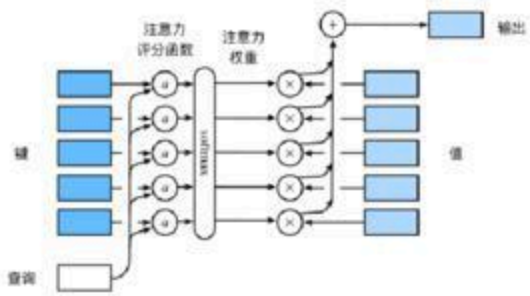
注意力池化：$f(x) = \sum_i \alpha(x, x_i) y_i$

查询$\mathbf{q}$和m个“键－值”对$(\mathbf{k}_m, \mathbf{v}_m)$

$$f(\mathbf{q}, (\mathbf{k}_1, \mathbf{v}_1), .., (\mathbf{k}_m, \mathbf{v}_m)) = \sum_i \alpha(\mathbf{q}, \mathbf{k}_i)\mathbf{v}_i$$

评分函数α

$$\alpha(\mathbf{q}, \mathbf{k}_i) = \mathrm{softmax}(a(\mathbf{q}, \mathbf{k}_i)) = \frac{\exp(a(\mathbf{q}, \mathbf{k}_i))}{\sum_j \exp(a(\mathbf{q}, \mathbf{k}_j}$$



- 评分函数决定注意力池化的最终输出

加性注意力

如果查询和键是**不同长度**的向量：转成相同维数

$$\alpha(\mathbf{q}, \mathbf{k}) = \mathbf{w}_v^T \tanh(\mathbf{W}_q\mathbf{q} + \mathbf{W}_k\mathbf{k})$$

- $\mathbf{W}_q \in \mathbb{R}^{h \times q}, \mathbf{W}_k \in \mathbb{R}^{h \times k}$：输出维数相同

加性注意力

如果查询和键是**不同长度**的向量：转成相同维数

$$\alpha(\mathbf{q}, \mathbf{k}) = \mathbf{w}_v^T \tanh(\mathbf{W}_q\mathbf{q} + \mathbf{W}_k\mathbf{k})$$

- $\mathbf{W}_q \in \mathbb{R}^{h\times q}, \mathbf{W}_k \in \mathbb{R}^{h\times k}$：输出维数相同
- 可以看成使用tanh激活的单隐藏层**感知机**，没有偏置项
  - 隐藏单元数h是超参数

加性注意力

如果查询和键是**不同长度**的向量：转成相同维数

$$\alpha(\mathbf{q}, \mathbf{k}) = \mathbf{w}_v^T \tanh(\mathbf{W}_q\mathbf{q} + \mathbf{W}_k\mathbf{k})$$

- $\mathbf{W}_q \in \mathbb{R}^{h\times q}, \mathbf{W}_k \in \mathbb{R}^{h\times k}$：输出维数相同
- 可以看成使用tanh激活的单隐藏层**感知机**，没有偏置项
  - 隐藏单元数h是超参数
- $\mathbf{w}_v$：可学习参数，用于区分隐状态的重要性

缩放点积注意力

如果查询和键是**相同长度**的向量：点积的计算效率更高

$$\alpha(\mathbf{q}, \mathbf{k}) = \mathbf{q}^T \mathbf{k} / \sqrt{d}$$

- 向量长度d：保证**归一化**后点积的方差为1
  - 假设“查询”、“键”的所有元素是随机变量，且满足0均值、单位方差
    - 向量点积：均值0、方差d

缩放点积注意力

如果查询和键是**相同长度**的向量：点积的计算效率更高

$$\alpha(\mathbf{q}, \mathbf{k}) = \mathbf{q}^T \mathbf{k} / \sqrt{d}$$

- 向量长度d：保证**归一化**后点积的方差为1
  - 假设“查询”、“键”的所有元素是随机变量，且满足0均值、单位方差
    - 向量点积：均值0、方差d

小批量提高效率：n个查询、m个键-值对

$$\text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\mathbf{T}}{\sqrt{d}}\right)\mathbf{V}$$

实验：注意力评分函数

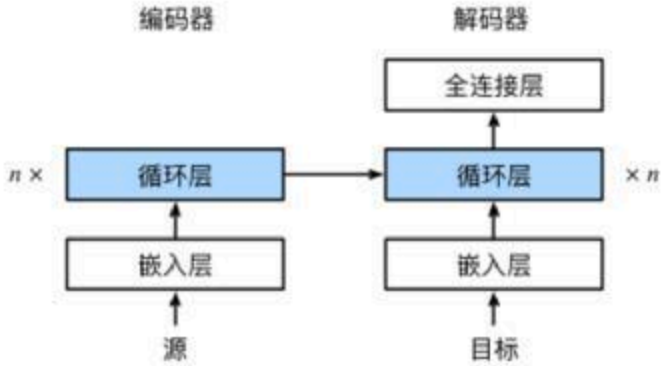
小结：注意力评分函数

- 注意力池化：“值”的加权平均
  - 注意力权重：“键”的评分函数值
- 不同评分函数带来不同注意力池化操作
  - “查询”、“键”长度不同：加性注意力
  - “查询”、“键”长度相同：缩放点积注意力

Bahdanau 注意力

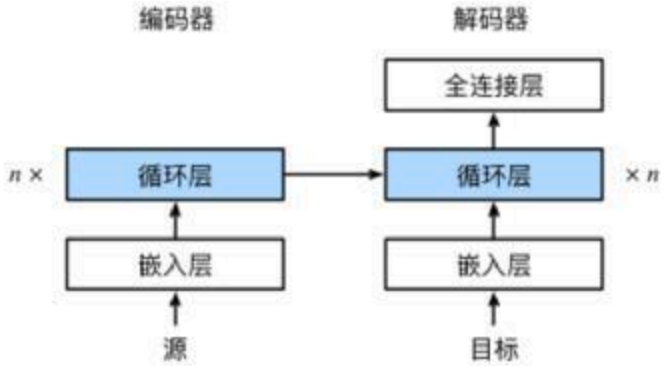
回顾：机器翻译

- 编码器：不输出、只**学习词嵌入**
  - 最后时间步隐状态：初始化解码器
- 解码器：根据上一输出**循环预测**



回顾：机器翻译

- 编码器：不输出、只**学习词嵌入**
  - 最后时间步隐状态：初始化解码器
- 解码器：根据上一输出**循环预测**

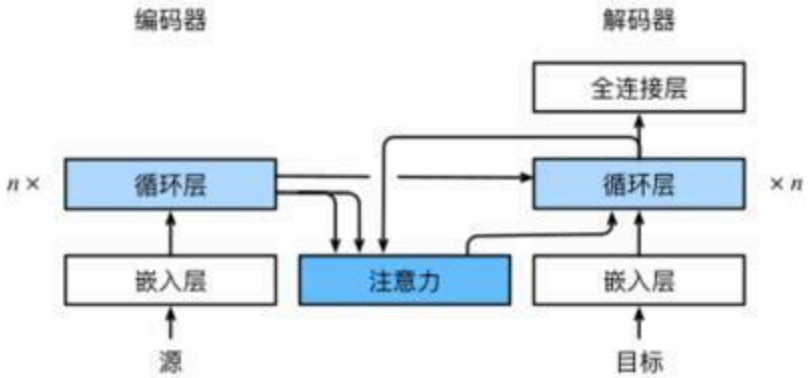


问题：并非所有源词元对当前预测都有用

- 但解码器使用**编码相同**的隐变量

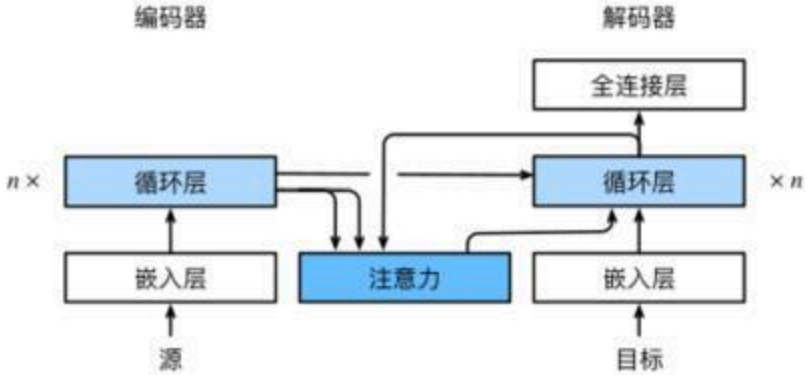
Bahdanau 注意力：模型

[Bahdanau 2014] 仅考虑**与当前预测相关**的输入子序列



- 编码器隐状态**既是“键”、也是“值”**
  - 注意观测注意力模块的三条输入路径

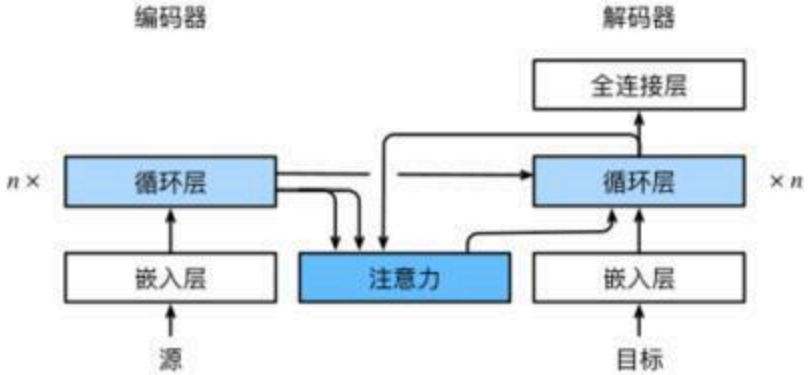
Bahdanau 注意力：解码器



初始化解码器状态需要以下输入

1. 编码器所有时间步的最终层隐状态：作为注意力的**键**、**值**
2. 上一时间步编码器的全层隐状态：用于**初始化**解码器隐状态
3. 编码器有效长度：**掩蔽**注意力池中填充词元

Bahdanau 注意力：解码器



初始化解码器状态需要以下输入

1. 编码器所有时间步的最终层隐状态：作为注意力的**键**、**值**
2. 上一时间步编码器的全层隐状态：用于**初始化**解码器隐状态
3. 编码器有效长度：**掩蔽**注意力池中填充词元

每个解码时间步：上一时间步的最终层隐状态用作**查询**

实验：Bahdanau 注意力

小结：Bahdanau 注意力

- 如果不是所有输入词元都相关：通过注意力机制筛选
- Bahdanau 注意力：解码器上一时间步的隐状态用于查询；编码器所有时间步的隐状态用作键、值

Review

本章内容

注意力提示。注意力池化。注意力评分函数。Bahdanau 注意力。

重点：两类注意力；Nadaraya-Watson 核回归、注意力池化；注意力权重、注意力评分函数；Bahdanau 注意力。

难点：Bahdanau 注意力模型的实现。

学习目标

- 理解两类注意力的区别（是否自主）与建模思路（全连接、自主提示）
- 了解 NW 核回归，并基于此理解两类（是否参数型）注意力池化的原理（“值”加权平均）、计算方法（“查询”、“键”之间的相似度）
- 理解注意力权重的意义（“键”的评分函数值），并基于此理解两类（加性、缩放点积）评分函数的原理、计算方法
- 理解 Bahdanau 注意力模型

问题

简述两类注意力的区别与建模思路。

简述两类注意力池化的原理、计算方法。

简述两类评分函数的原理、计算方法。

简述 Bahdanau 注意力模型。