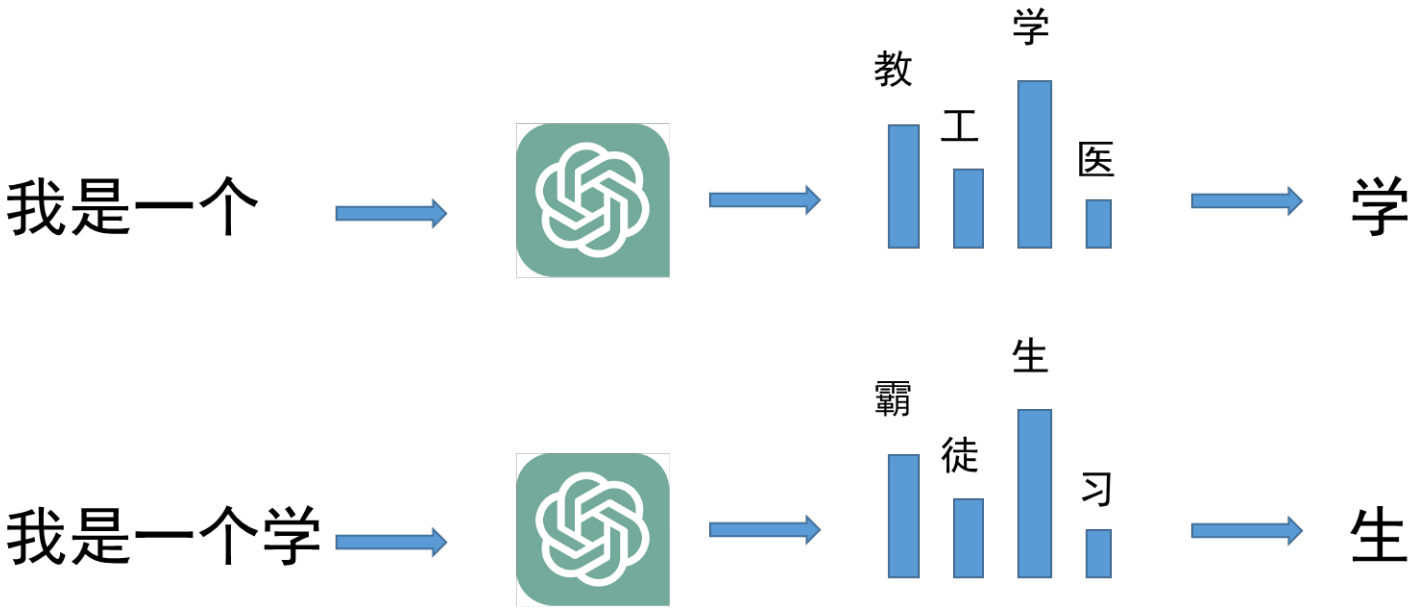
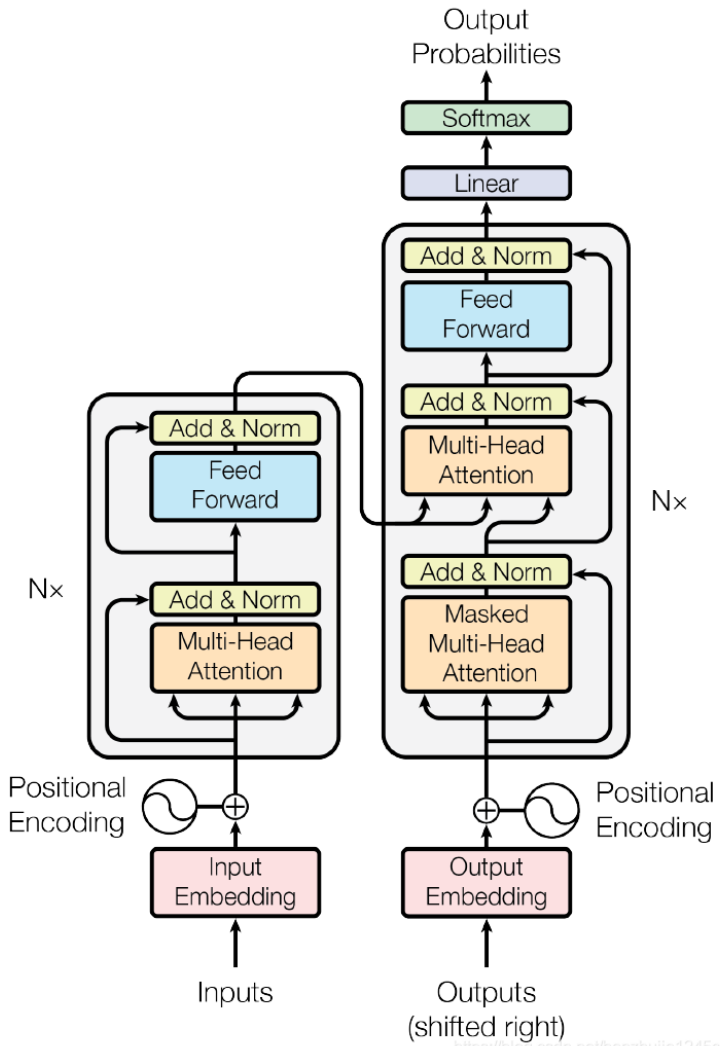
Transformer——大模型的基石

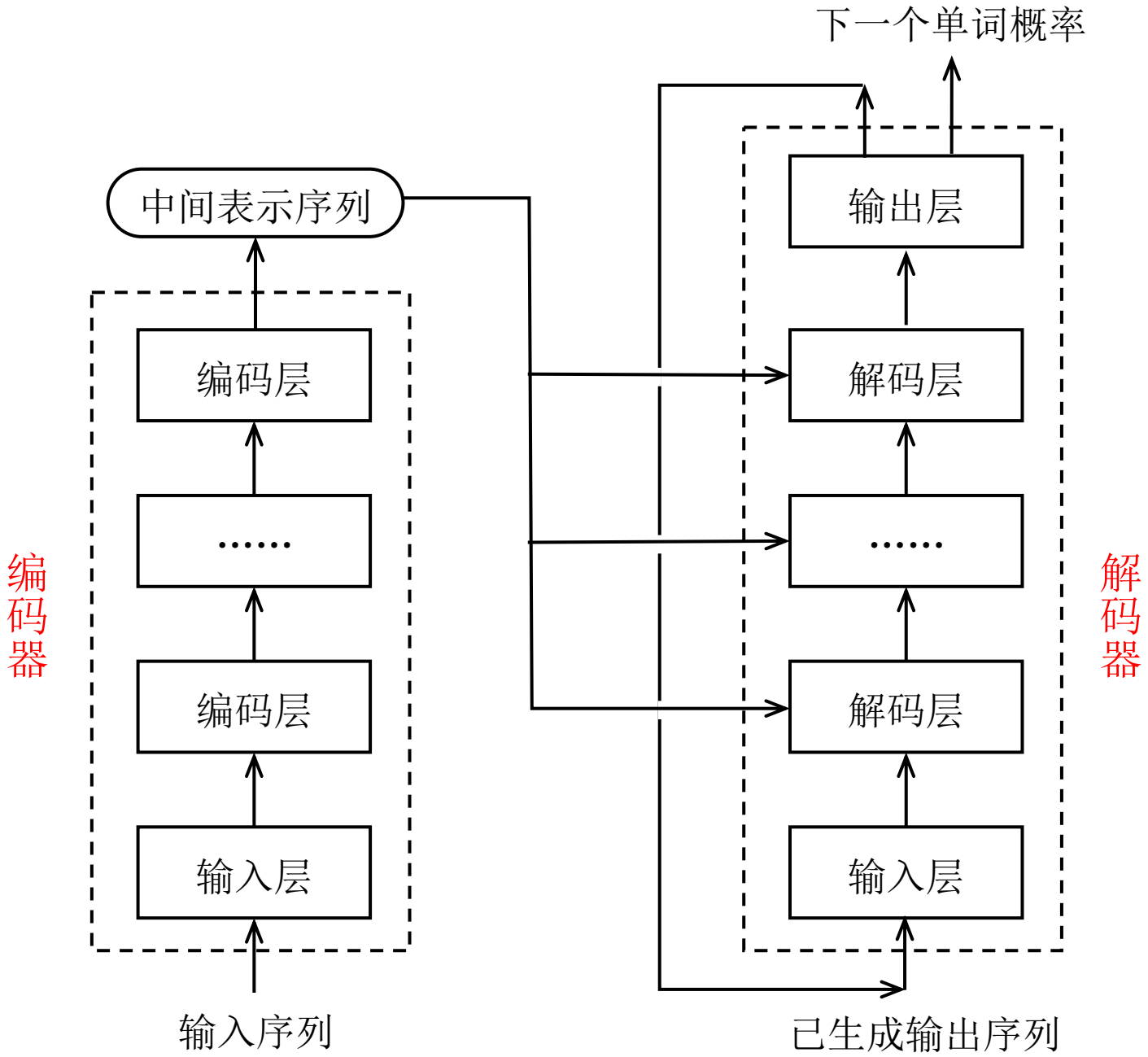
GPT：生成式预训练转换模型

- 文字接龙

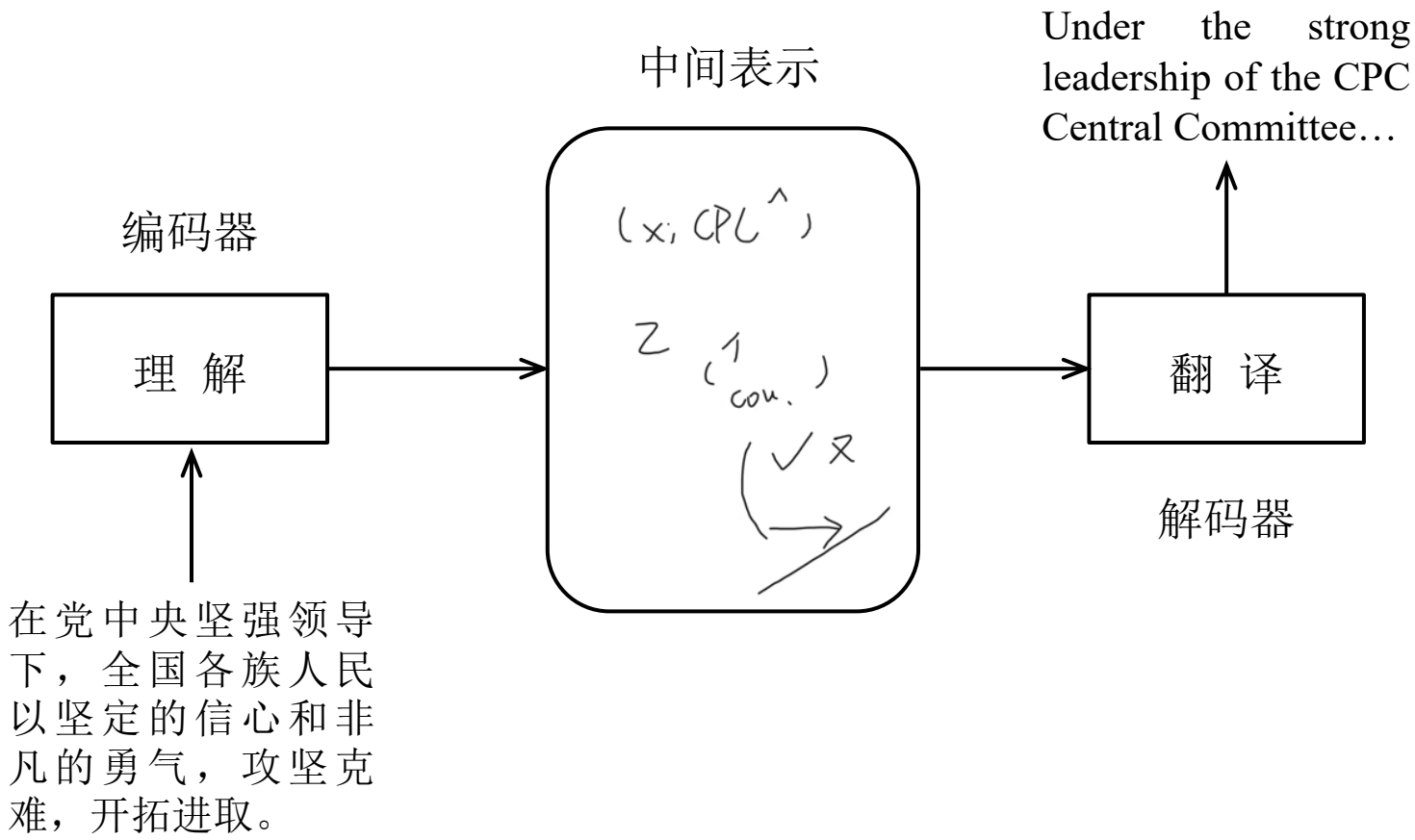


Transformer：转换模型

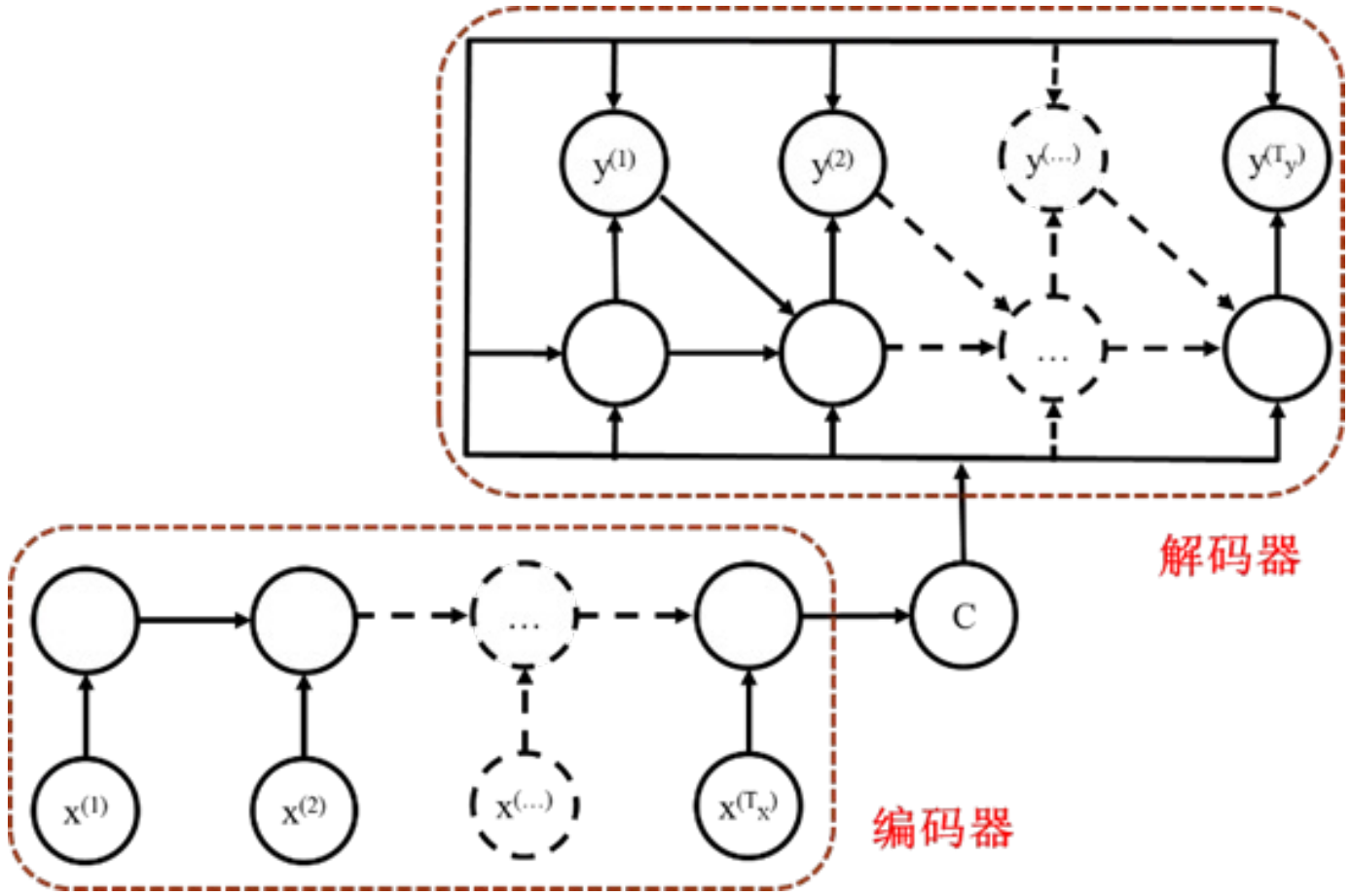



下一个单词概率
中间表示序列
输出层
编码层
解码层
……
……
编码层
解码层
输入层
输入层
编码器
解码器
输入序列
已生成输出序列

编码器-解码器模型


中间表示
Under the strong leadership of the CPC Central Committee…
编码器
理 解
翻 译
解码器
在党中央坚强领导下，全国各族人民以坚定的信心和非凡的勇气，攻坚克难，开拓进取。

编码器-解码器模型

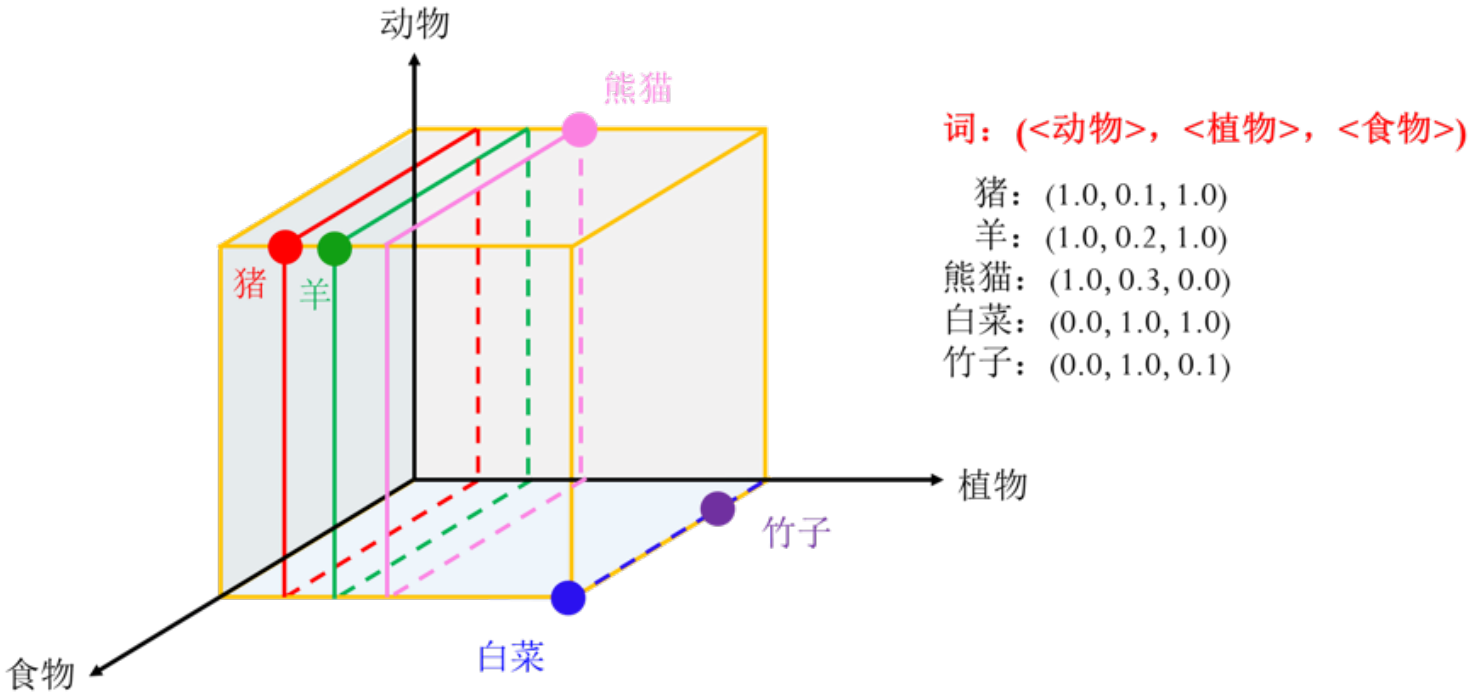


关键技术

- 两个关键技术
  - 词向量的动态表示
  - 注意力机制

词向量：静态与动态

- 静态性
- 词的多义性



一个词可以用其上下文表示

“我吃了一个非常美味的苹果”

“我用苹果跟朋友联系”

注意力机制

键-值数据库查询

- 键-值对数据
  - $\{(k_1, v_1), (k_2, v_2), \ldots, (k_n, v_n)\}$
- 查询q

- 例：
  - （年龄i-收入i）
  - 查询：中年人的收入

  - 按照“年龄”与“中年人”的相似程度计算加权平均值

注意力机制

$$v = \sum_{i=1}^{n} \alpha(q, k_i) v_i$$

$$\sum_{i=1}^{n} \alpha(q, k_i) = 1$$

$$\alpha(q, k_i) = \text{softmax}\big(sim(q, k_i)\big) = \frac{e^{sim(q,k_i)}}{\sum_{j=1}^{n} e^{sim(q,k_j)}}$$

$sim(q, k_i)$：表示q与k_i的相似性

$\alpha(q, k_i)$：注意力

注意力机制

- q, k_i, v_i均用维度为d的行向量表示
- 键矩阵：$K = \begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}_{n \cdot d}$　值矩阵：$V = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}_{n \cdot d}$　查询矩阵：$Q = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_m \end{bmatrix}_{m \cdot d}$
- 相似性：

$$sim(q_i, k_j) = \frac{q_i k_j^T}{\sqrt{d}}$$

标量

- 注意力：

$$att(q_i, K, V) = softmax\left(\frac{q_i K^T}{\sqrt{d}}\right) \cdot V$$

d维行向量

对矩阵的每一行做softmax

$$att(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V$$

$m \cdot d$矩阵

自注意力机制

- 长度为n的输入序列x：$x_1, x_2, \ldots, x_n$
- 序列矩阵：

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}_{n \cdot d}$$

- 自注意力机制：

$$att(Q,K,V) = att(X,X,X) = softmax\left(\frac{XX^T}{\sqrt{d}}\right) \cdot X_{n \cdot d}$$

- $att(Q,K,V)$的第i行为序列中第i个单词的动态向量表示
  - 列数为向量的维度d
  - 行数为序列长度n

多头注意力机制

- 分别对Q、K、V线性变换后再计算机注意力

$$U_i = att\left(QW_q^{(i)}, KW_k^{(i)}, VW_v^{(i)}\right), i = 1,2,\dots,h$$

$$U = [U_1; U_2; \dots; U_h]_{\,n\cdot(d\cdot h)}$$

- 多头注意力机制

$$multi\_att(Q,K,V) = U \cdot W_o$$

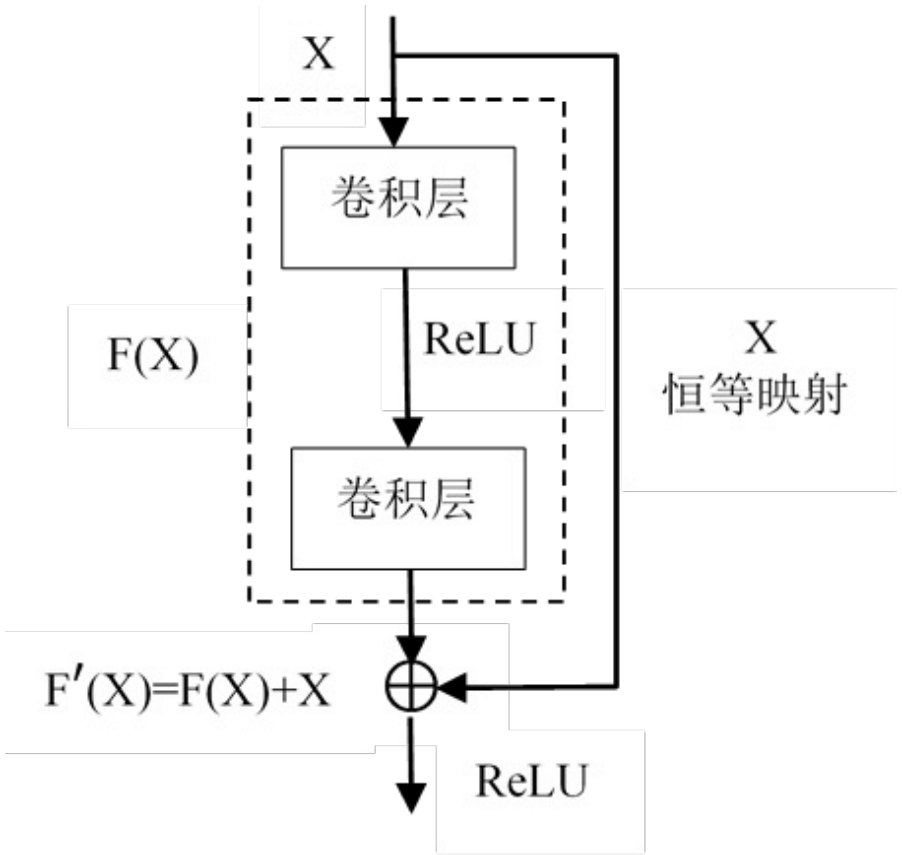
W_o: $(d\cdot h)\cdot d$

$multi\_att(Q,K,V)$: $n\cdot d$

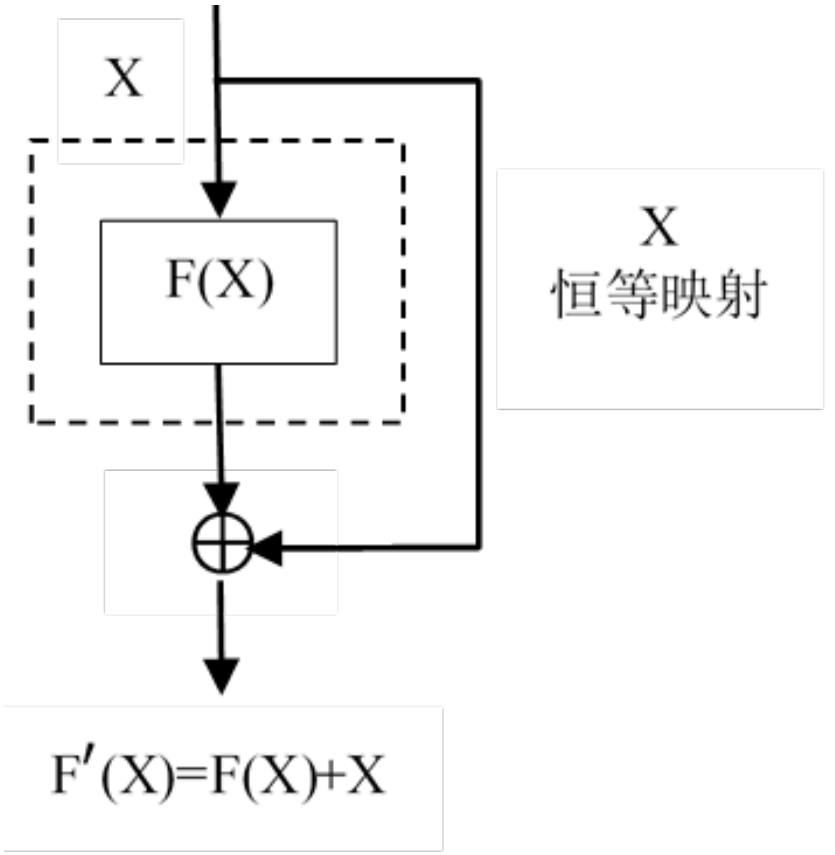
- $multi_att(Q,K,V)$的第i行为序列中第i个单词的动态向量表示

残差连接

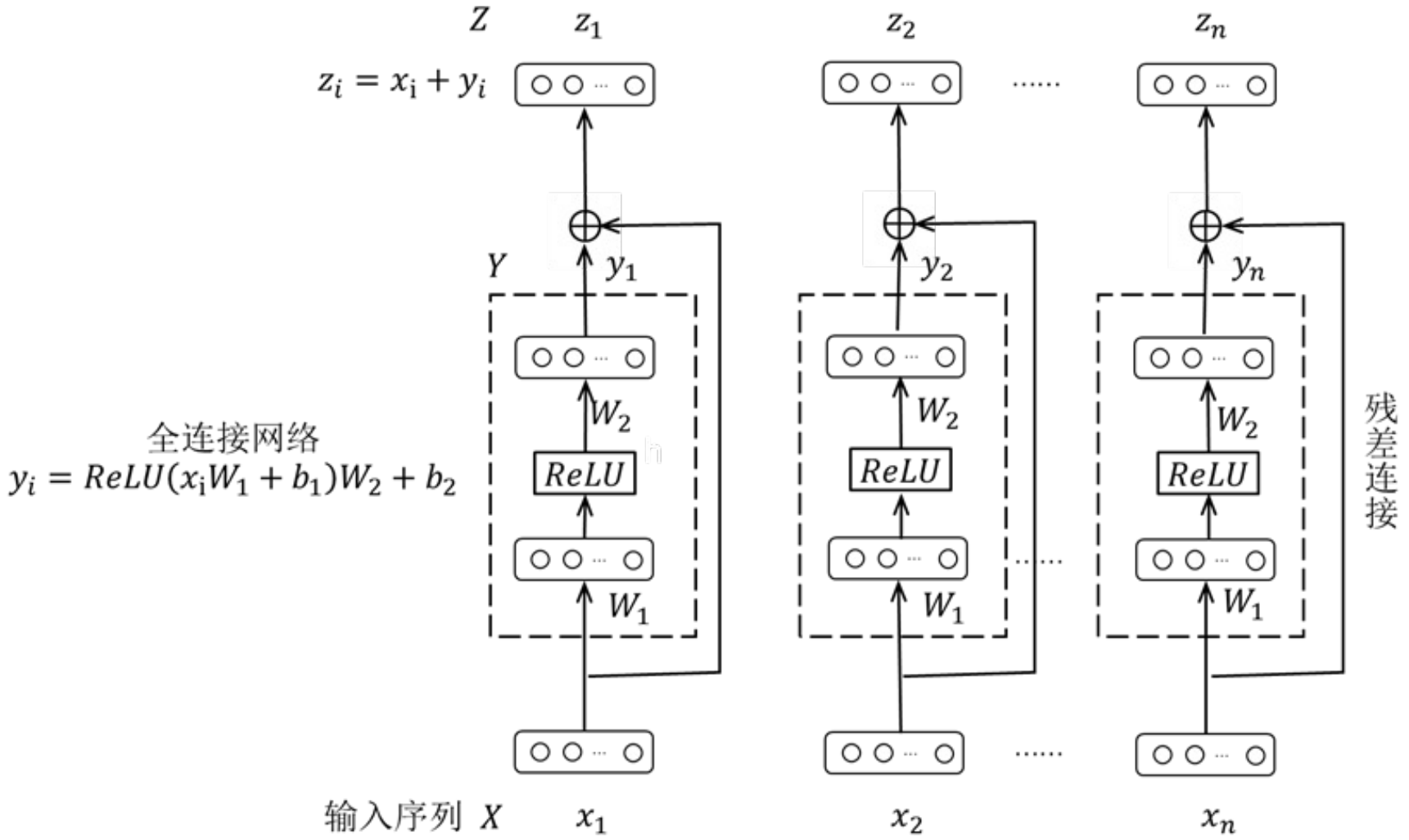
残差网络



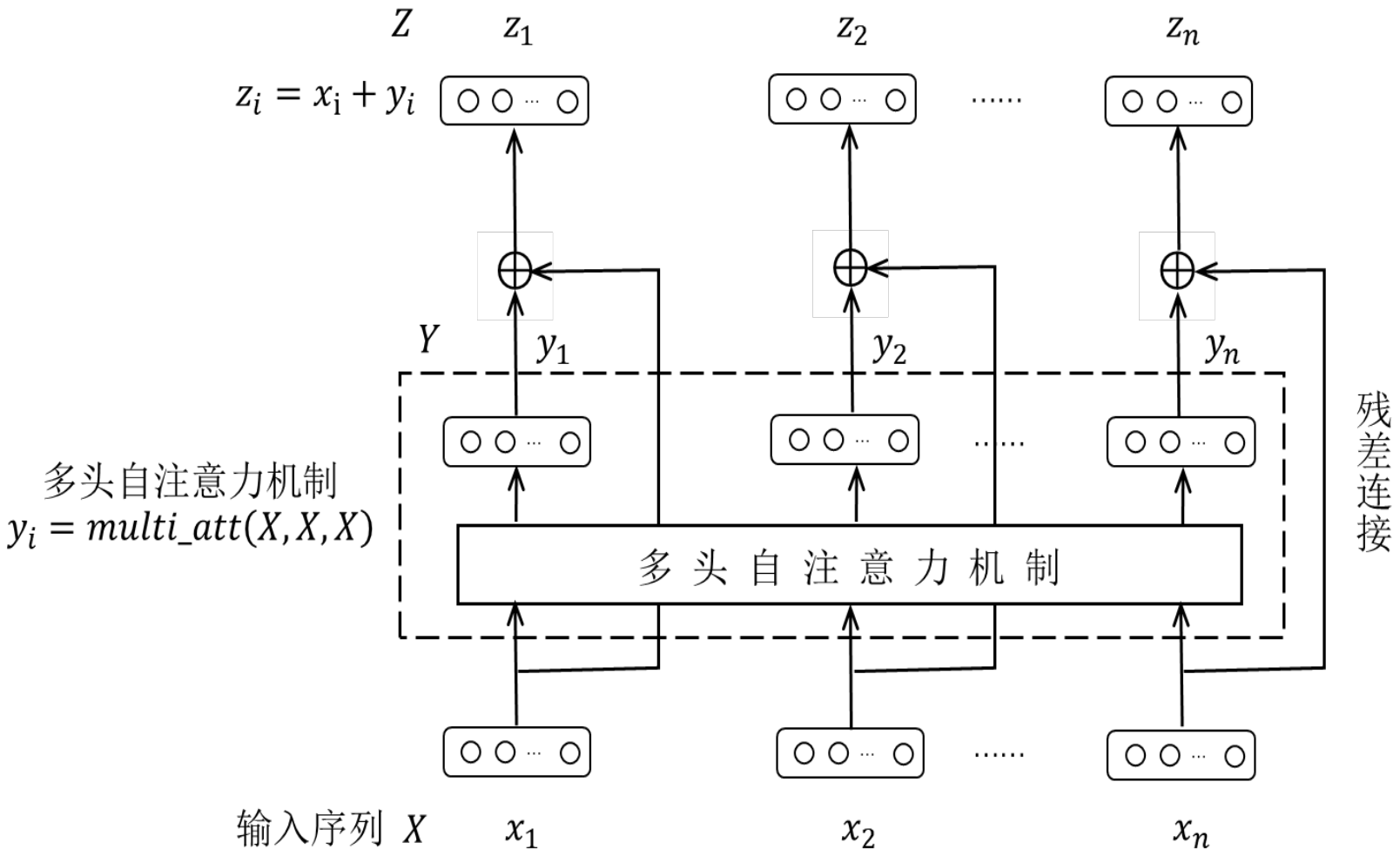
一般形式



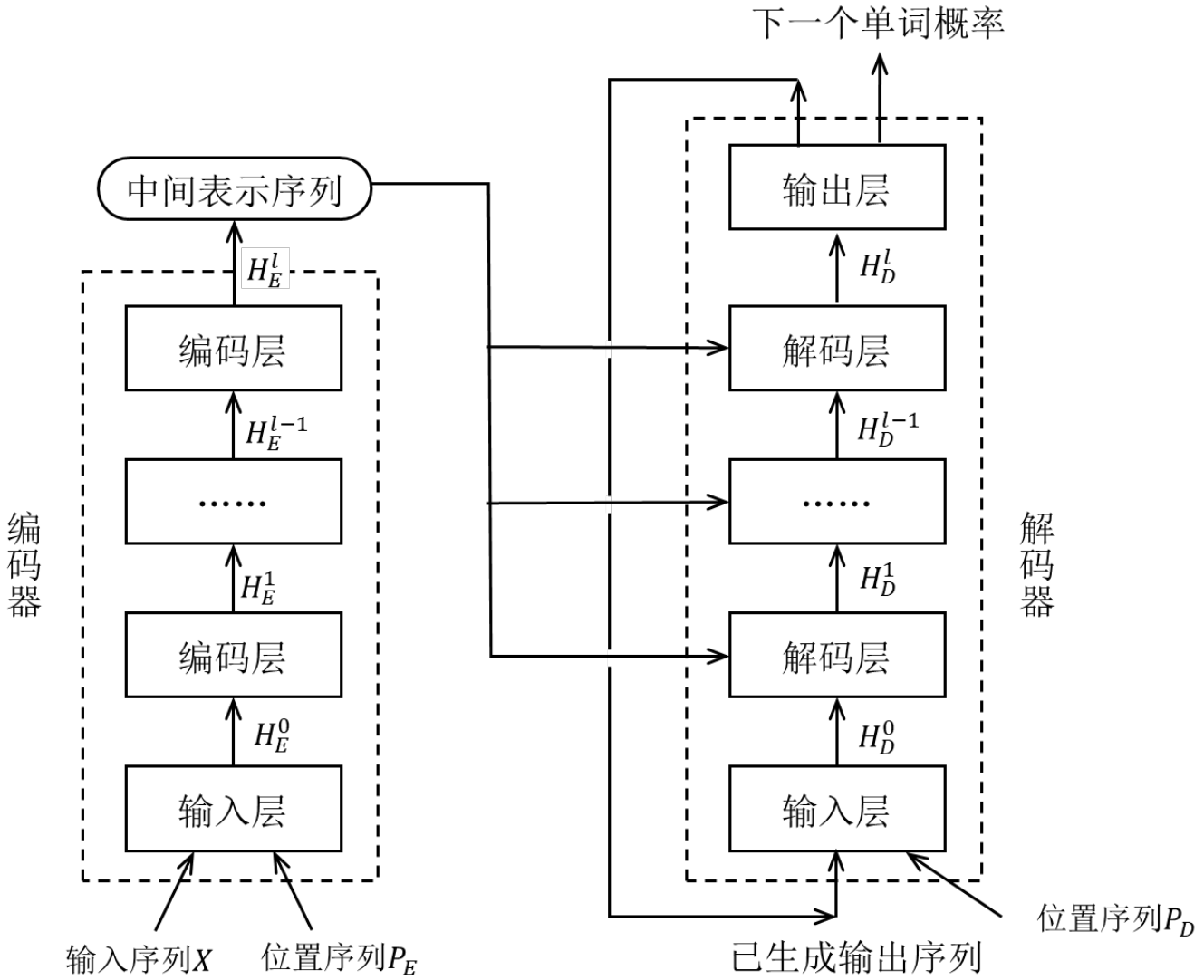
全连接网络+残差连接


Z
z_1
z_2
z_n
$z_i = x_i + y_i$
.....
Y
y_1
y_2
y_n
W_2
ReLU
W_1
全连接网络
$y_i = ReLU(x_i W_1 + b_1)W_2 + b_2$
残差连接
输入序列 X
x_1
x_2
x_n

多头注意力机制+残差连接

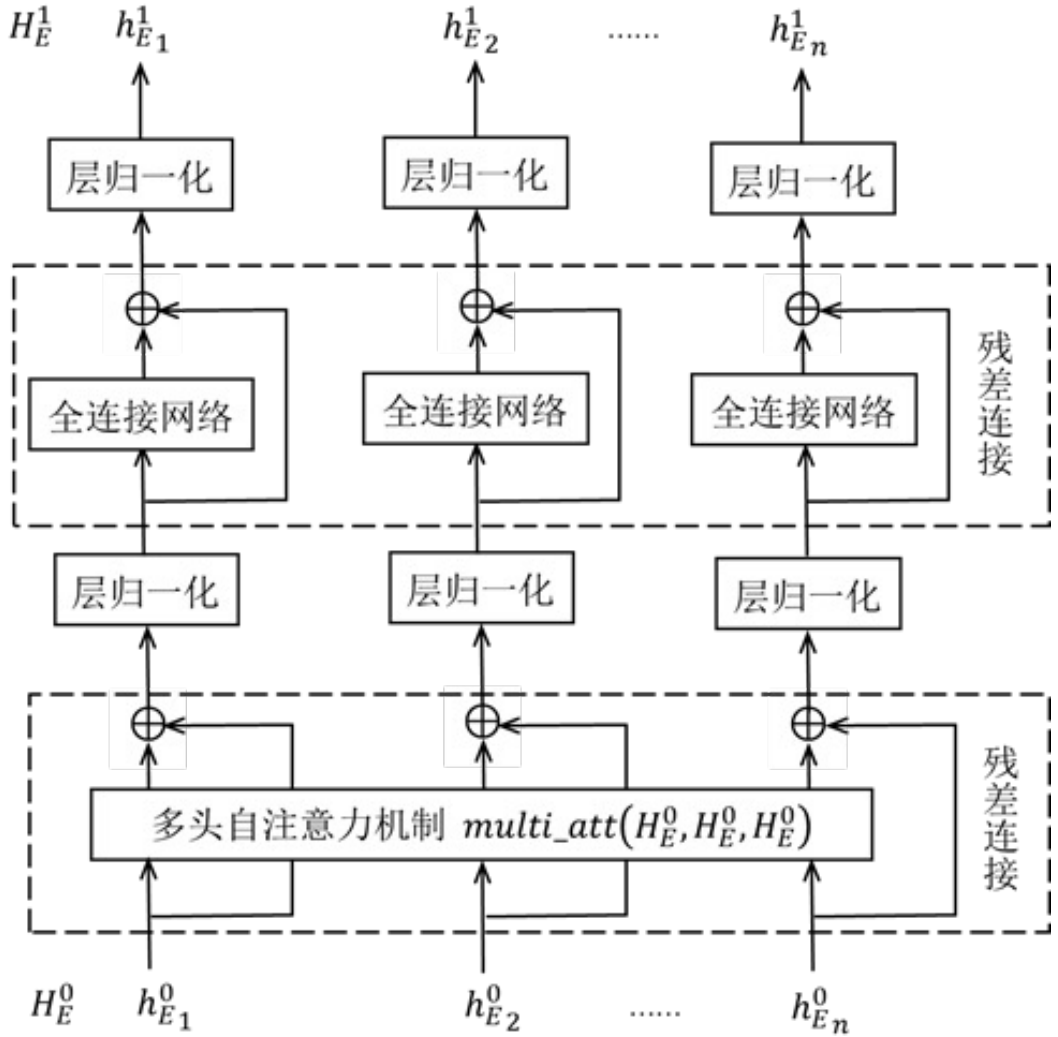


Transformer



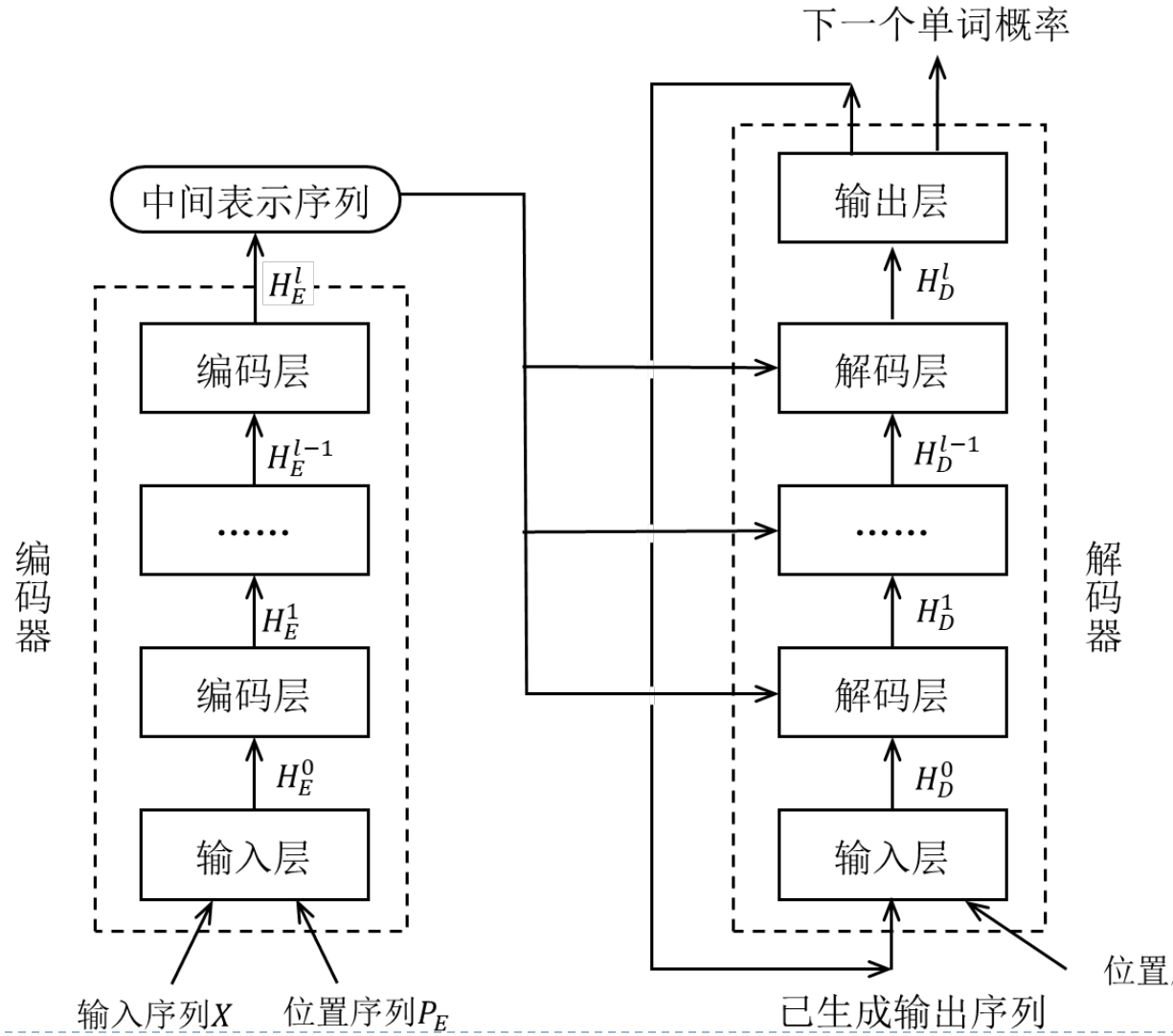
$$H_E^0 = X + P_E$$

编码层


H_E^1
$h_{E_1}^1$
$h_{E_2}^1$
.....
$h_{E_n}^1$
层归一化
层归一化
层归一化
全连接网络
全连接网络
全连接网络
残差连接
层归一化
层归一化
层归一化
多头自注意力机制 $multi\_att(H_E^0, H_E^0, H_E^0)$
残差连接
H_E^0
$h_{E_1}^0$
$h_{E_2}^0$
.....
$h_{E_n}^0$

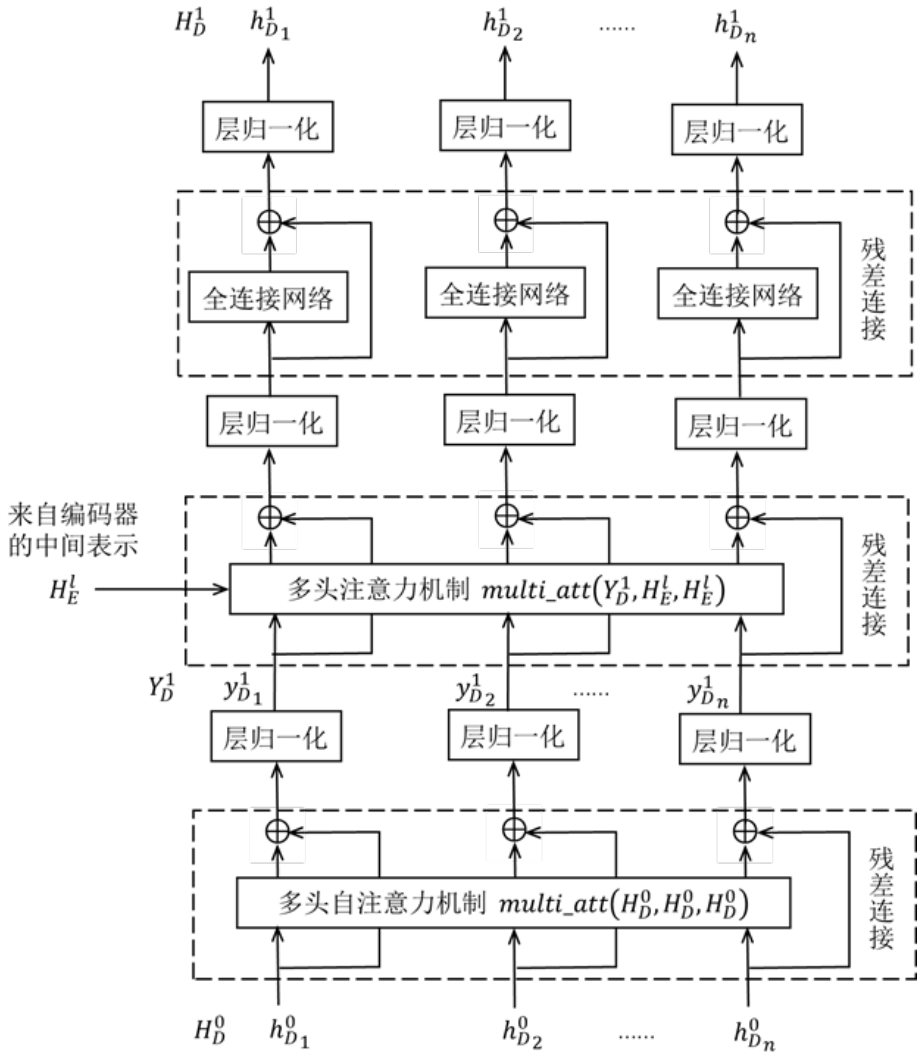
$$H_E^0 = X + P_E$$

Transformer



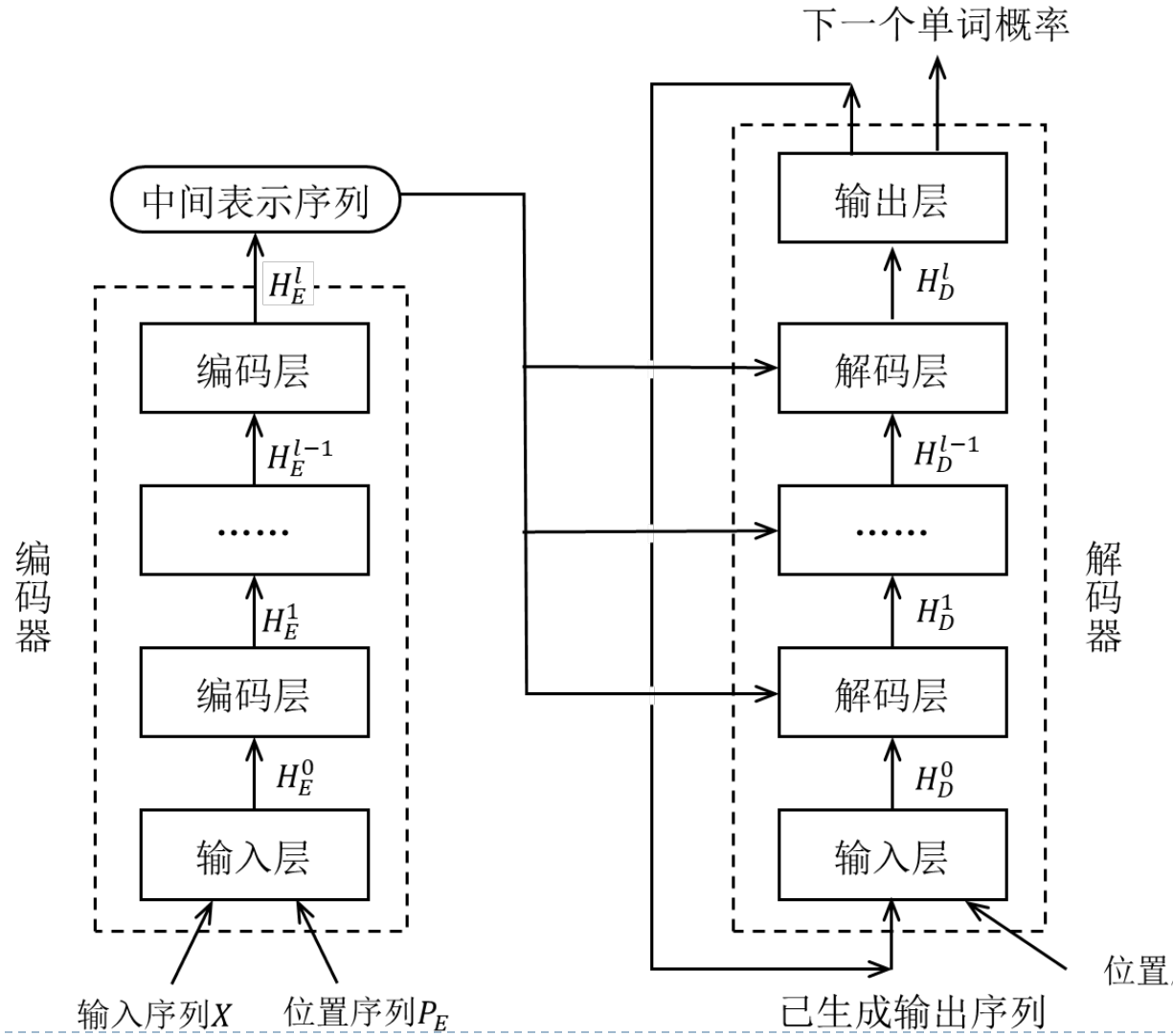
$$H_D^0 = E + P_D$$

解码层


H_D^1 $h_{D_1}^1$ $h_{D_2}^1$ ..... $h_{D_n}^1$
层归一化
全连接网络
残差连接
来自编码器的中间表示
H_E^l
多头注意力机制 $multi\_att(Y_D^1, H_E^l, H_E^l)$
Y_D^1 $y_{D_1}^1$ $y_{D_2}^1$ ..... $y_{D_n}^1$
多头自注意力机制 $multi\_att(H_D^0, H_D^0, H_D^0)$
H_D^0 $h_{D_1}^0$ $h_{D_2}^0$ ..... $h_{D_n}^0$

$$H_D^0 = E + P_D$$

Transformer



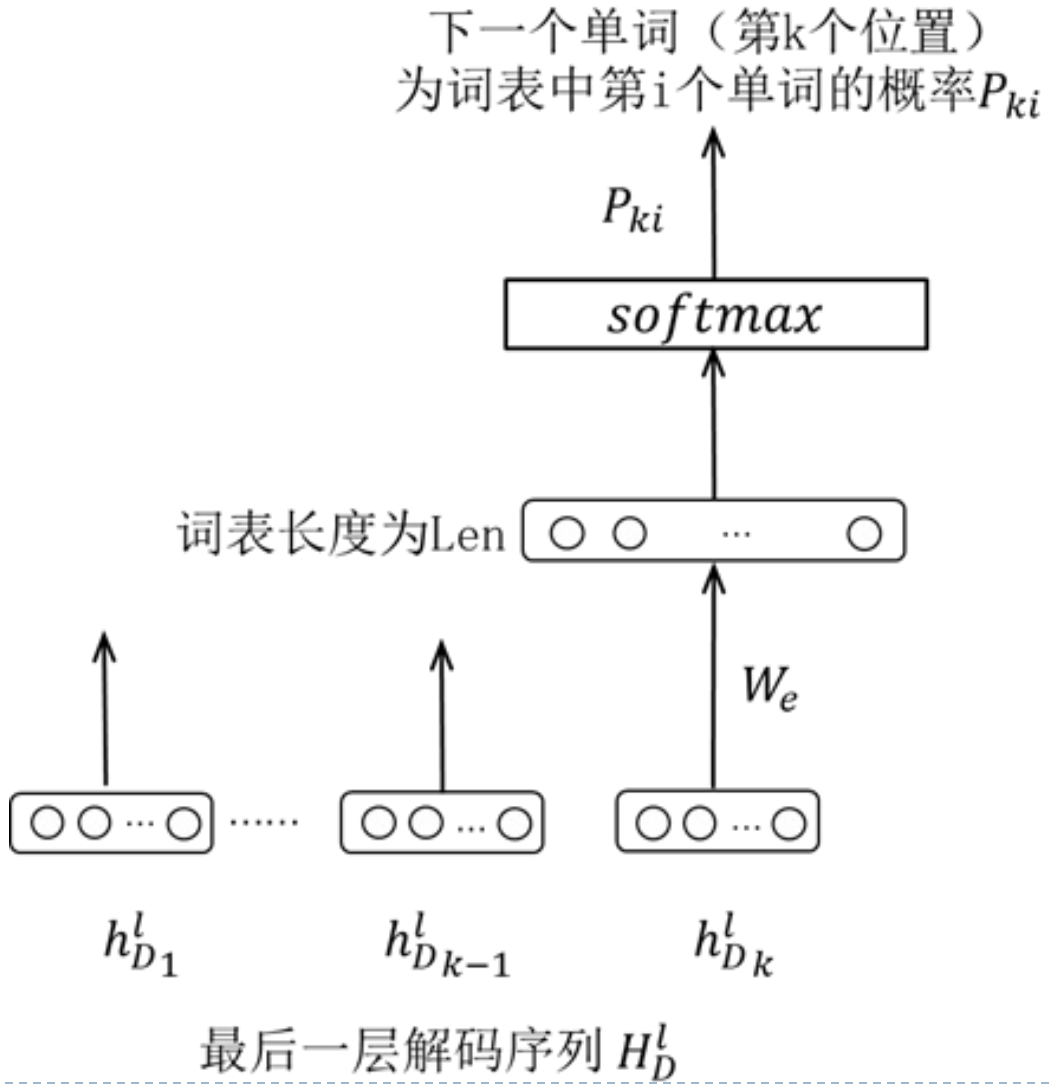
$$H_D^0 = E + P_D$$

输出层

W_e：第i行为词表中第i个单词的词向量

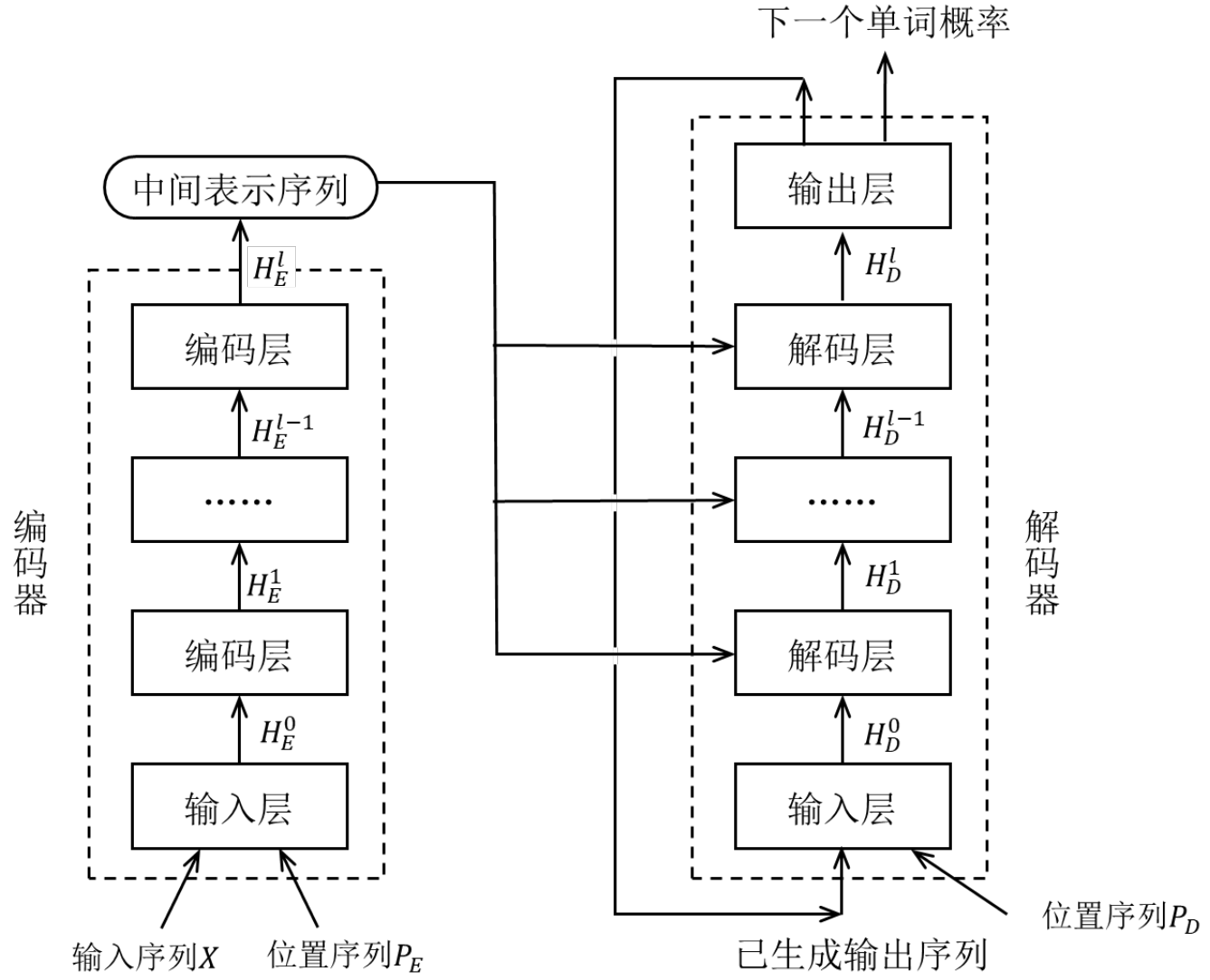
$$P_k = softmax\left(h_{D_k}^l \cdot {W_e}^T\right)$$

p_k的维度为词表长度，其第i个元素p_{ki}为第k个位置为词汇表中第i个单词的概率



输出序列

- 编码器
  - 运行一次
- 解码器
  - 自回归
  - 运行m+l次
  - m为输出序列的长度
  - 结束符



Transformer的训练

- 最大化似然函数

$$\max_{\theta} \prod_{w \epsilon C} p(w = k | context(w), \theta)$$

- 损失函数

$$L(\theta) = -\sum_{w \epsilon C} \ln\big(p(w = k | context(w), \theta)\big)$$

Transformer的应用

- GPT
  - 只使用解码器
  - 自回归
- BERT
  - 只用编码器
  - 双向（利用前后词预测中间词）