NNLM(Neural Network Language Models)

i-th output = P(Wt = i | contex)

softmax

most computation here

tanh

$C(W_{t-n+1})$ $C(W_{t-2})$ $C(W_{t-1})$

Table look-up in C

matrix C

shared parameters across words

Index for W_{t-n+1} Index for W_{t-2} Index for W_{t-1}

2 5 7 8 1 0 1 7

=

m列

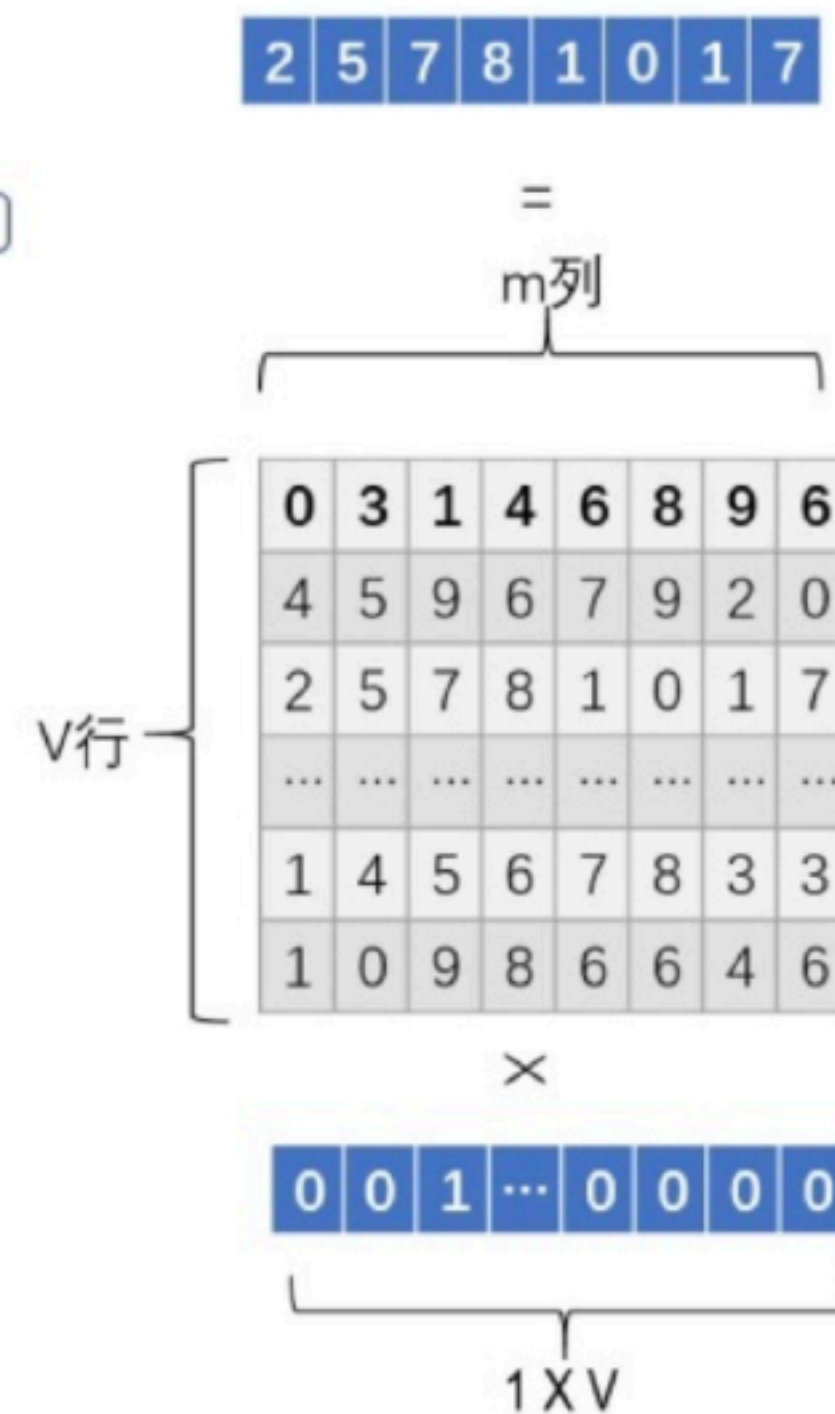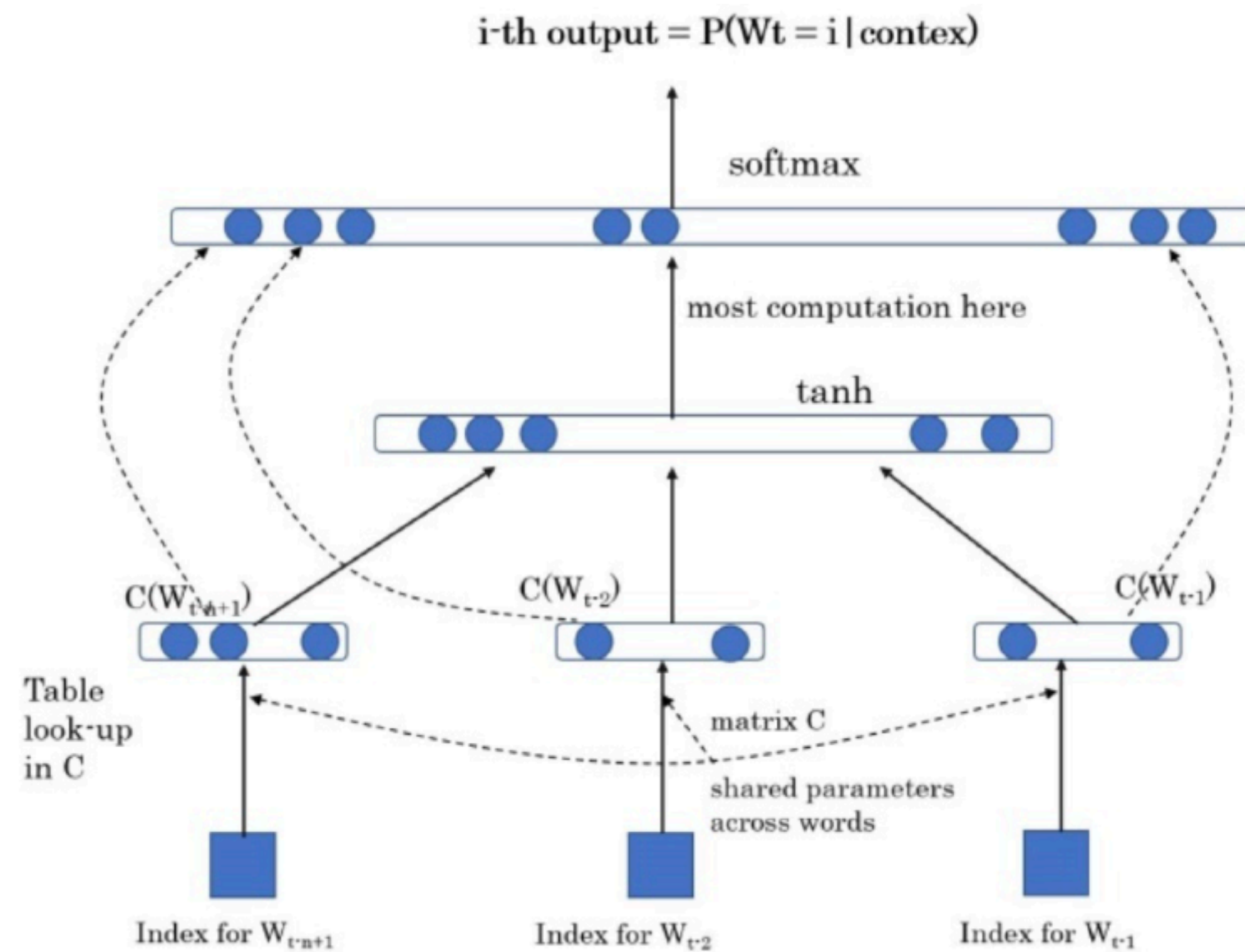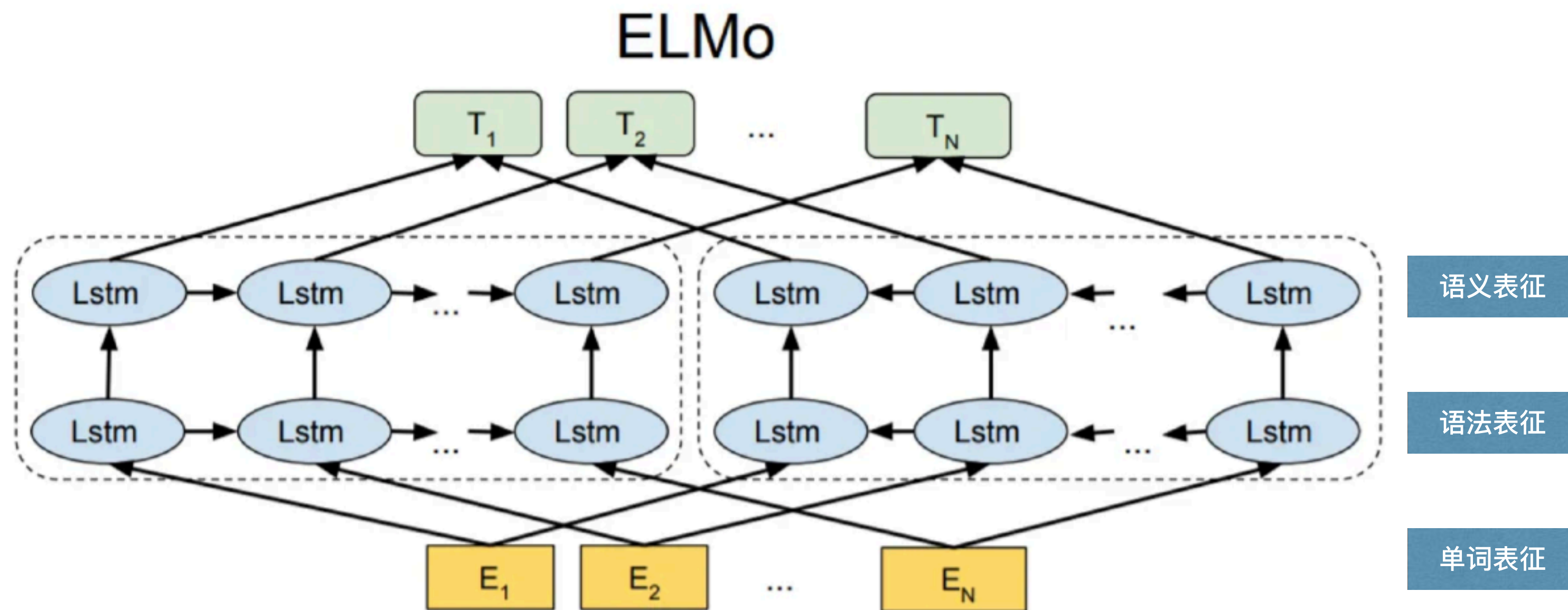| 0 | 3 | 1 | 4 | 6 | 8 | 9 | 6 |
4	5	9	6	7	9	2	0
2	5	7	8	1	0	1	7
…	…	…	…	…	…	…	…
1	4	5	6	7	8	3	3
1	0	9	8	6	6	4	6

V行

×

0 0 1 … 0 0 0 0

1 X V

Input Layer: word indices
Projection Layer: word embedding
Hidden Layer: vector calculation
Output Layer: the probability or the next word

Word Embedding

$$y = \text{softmax}(b + Wx + U \tanh(d + Hx))$$

ELMo(Embeddings from Language Models)

ELMo

https://arxiv.org/pdf/1802.05365.pdf

使用双向LSTM将要预测的词汇上下构建起来

整个网络分三层：

Word-Embedding Layer
Contextualized Layer
Output Layer

其中在Contextualized Layer中，以被预测词汇

为中心，分为:

Context-Before
Context-After

经过网络处理后的每个单词都会得到三个

Embedding

最底层的单词Embedding

上一层的语法Embedding

再上一层的语义Embedding