深度学习与机器翻译

前沿技术讲习班

Advanced Technology Tutorial

统计机器翻译中的深度学习

张家俊

中国科学院自动化研究所

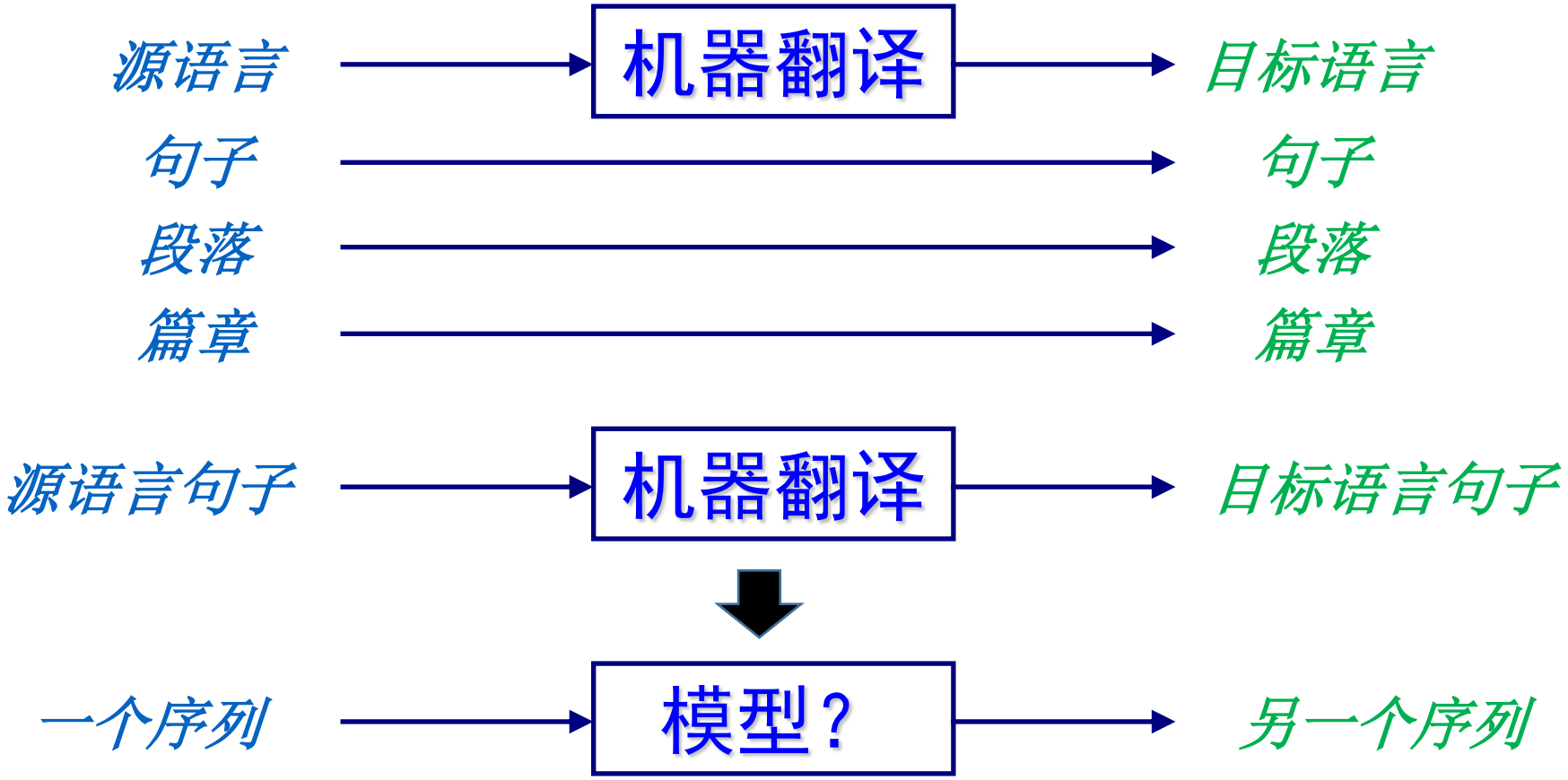
www.nlpr.ia.ac.cn/cip/jjzhang.htm

jjzhang@nlpr.ia.ac.cn

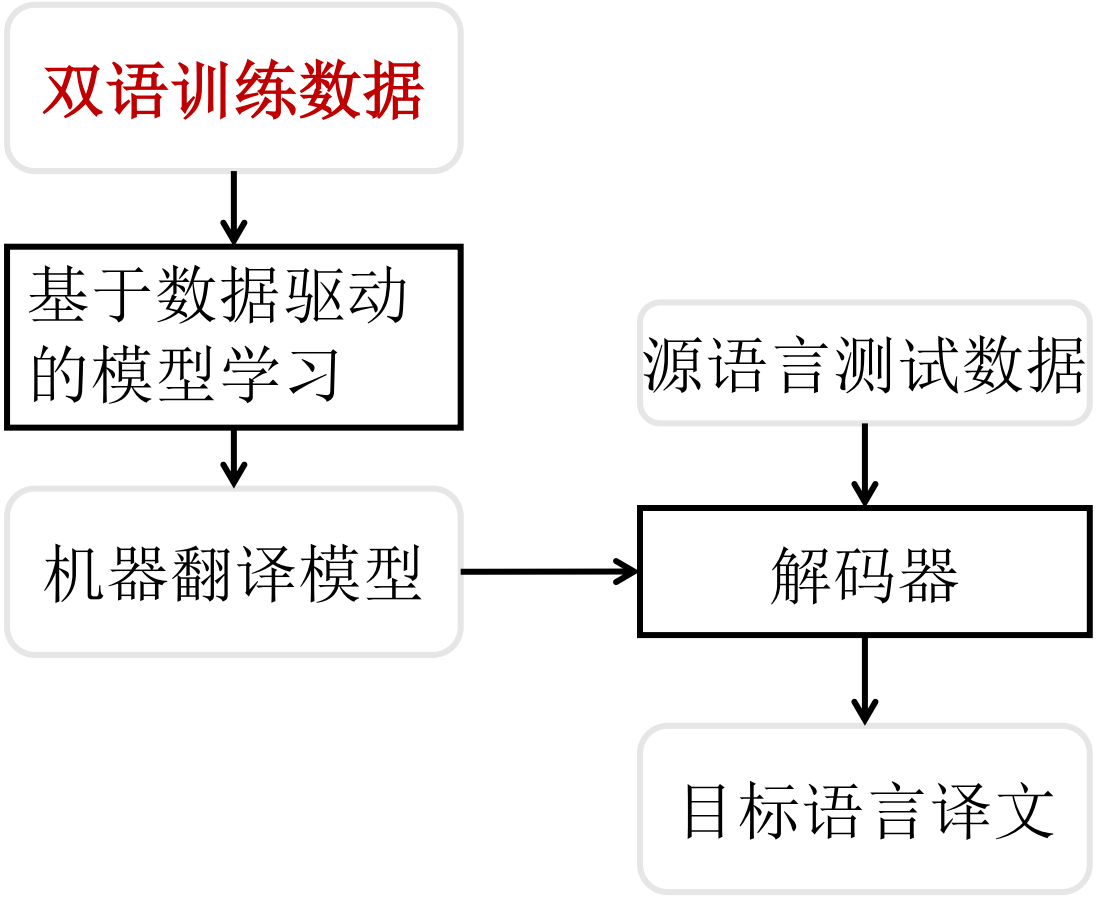
机器翻译

今天烟台天气不错

The weather is fine in Yantai today



统计机器翻译



双语训练数据

人类 共 有 二十三 对 染色体 。	澳洲 重新 开放 驻 马尼拉 大使馆
humans have a total of 23 pairs of chromosomes .	australia reopens embassy in manila
中国 大陆 手机 用户 成长 将 减缓	外交 人员 搭乘 第五 架 飞机 返国
growth of phone users in mainland china to slow	diplomatic staff take the fifth plane home
驻 南韩 美军 三千人 奉命 冻结 调防	姚明 感慨 NBA 的 偶像 来 得 太 快
us freezes transfer of 3,000 troops in south korea	yao ming feels nba stardom comes too fast
… …	… …

统计机器翻译-生成模型

源语言句子： $S = s_1^m = s_1 s_2 \cdots s_m$

目标语言句子： $T = t_1^l = t_1 t_2 \cdots t_l$

贝叶斯公式： $P(T|S) = \dfrac{P(T) \times P(S|T)}{P(S)}$

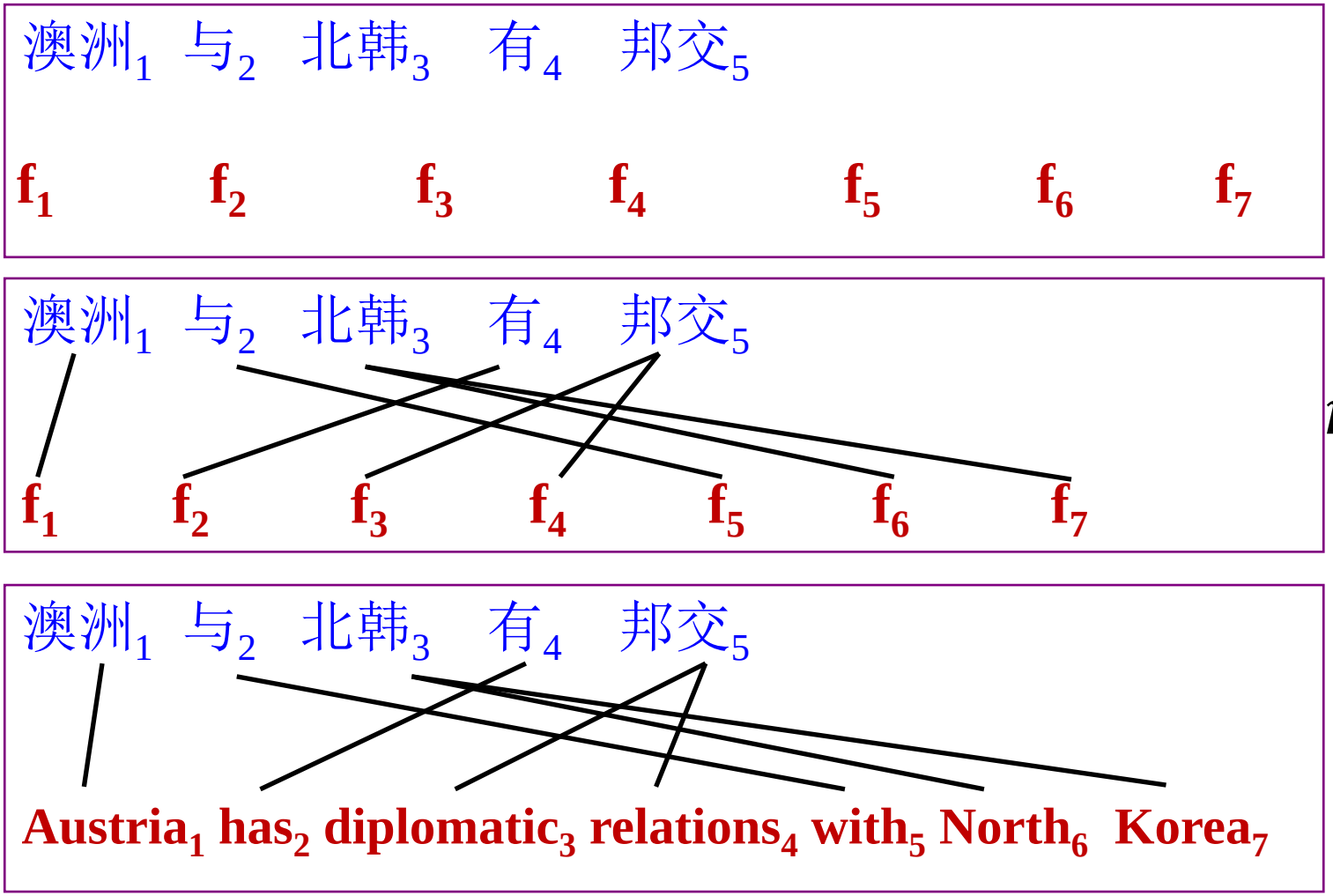
$$T' = \underset{T}{\operatorname{argmax}}\, P(T) \times P(S|T)$$

[Brown et al., 1990, 1993]

语言模型
Language model, LM

翻译模型
Translation model, TM

统计机器翻译-生成模型



IBM Model-2

统计机器翻译-判别式模型

[Och, 2002]

$$T' = \arg\max_T P(T|S)$$

$$= \arg\max_T \frac{P(T) \times P(S|T)}{P(S)}$$

$$= \arg\max_T P(T) \times P(S|T)$$

生成式模型

$$T' = \arg\max_T P(T) \times P(T|S)$$

?

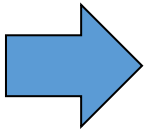
翻译质量 ≈ 翻译质量

统计机器翻译-判别式模型

$$T' = \underset{T}{\operatorname{argmax}} P(T) \times P(S|T)$$

$$T' = \underset{T}{\operatorname{argmax}} P(T) \times P(T|S)$$

翻译质量



$$T' = \underset{T}{\operatorname{argmax}} P(T) \times P(S|T) \times P(T|S)$$

<

翻译质量

统计机器翻译-判别式模型

$$T' = \underset{T}{\operatorname{argmax}} P(T|S) = \underset{T}{\operatorname{argmax}} \frac{exp\{\sum_1^M \lambda_m h_m(T,S)\}}{\sum_{T^*} exp\{\sum_1^M \lambda_m h_m(T^*,S)\}}$$

$$= \underset{T}{\operatorname{argmax}} \left\{ \sum_1^M \lambda_m h_m(T,S) \right\}$$

$$h_1(T,S) = logP(T)$$

$$h_2(T,S) = logP(S|T)$$

$$\lambda_1 = \lambda_2 = 1$$

$$T' = \underset{T}{\operatorname{argmax}} P(T|S) = \underset{T}{\operatorname{argmax}} \left\{ \sum_1^M \lambda_m h_m(T,S) \right\}$$

$$= \underset{T}{\operatorname{argmax}} \{logP(T) + logP(S|T)\}$$

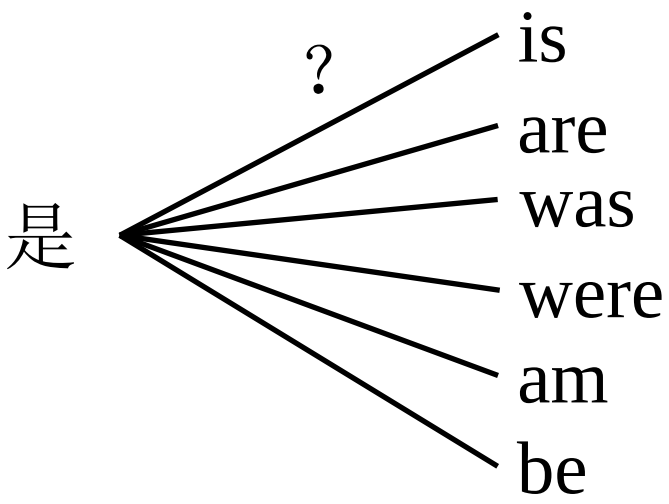
$$= \underset{T}{\operatorname{argmax}} \{log(P(T) \times P(S|T))\}$$

$$= \underset{T}{\operatorname{argmax}} \{P(T) \times P(S|T)\}$$

基于短语的统计机器翻译

- 基于词的翻译模型的问题：
  - 很难处理词义消岐问题
  - 很难处理一对多、多对一和多对多的翻译问题

澳洲 是 与 北韩 有 邦交 的 少数 国家 之一



基于短语的统计机器翻译

- 基于词的翻译模型的问题：
  - 很难处理词义消岐问题
  - 很难处理一对多、多对一和多对多的翻译问题

澳洲 是 与 北韩 有 邦交 的 少数 国家 之一

北韩 $\xrightarrow{?}$ North Korea

邦交 $\xrightarrow{?}$ the diplomatic relations

基于短语的统计机器翻译

- **基于短语的统计机器翻译：**

澳洲 是 与 北韩 有 邦交 的 少数 国家 之一

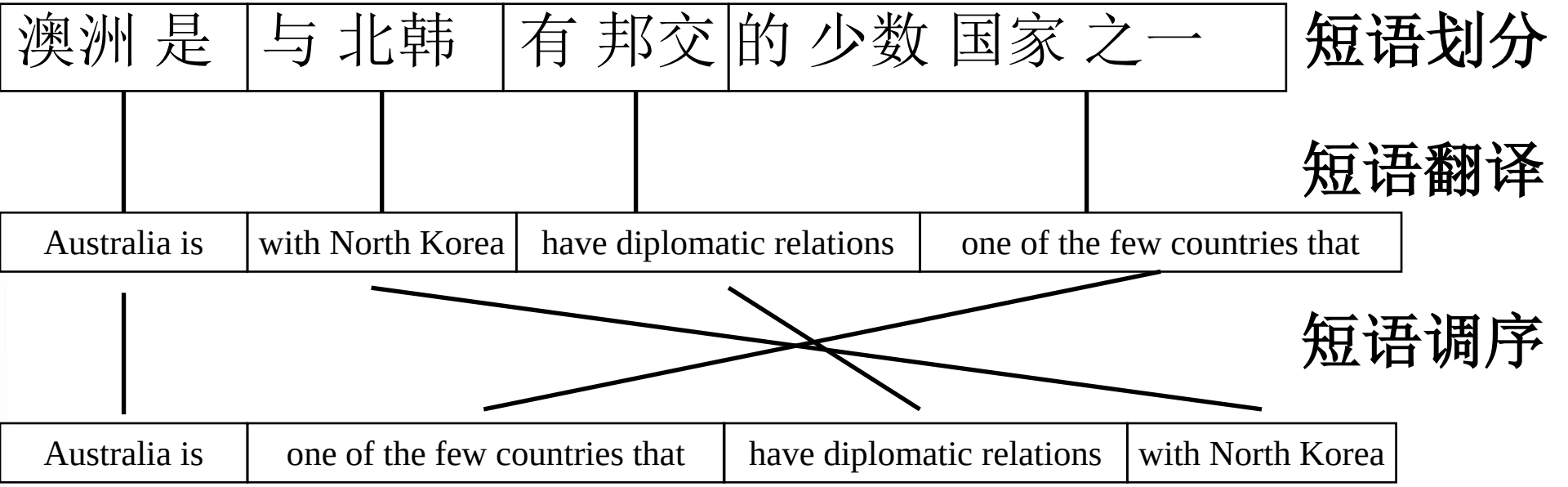
(澳洲 是, Australia is)

(与 北韩, with North Korea)

(有 邦交, have the diplomatic relations)

(的 少数 国家 之一, one of the few countries that)

基于短语的统计机器翻译



基于短语的统计机器翻译

[Koehn, 2003]

短语：连续的词串 （非句法意义）

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \boldsymbol{P}(S_1^K|\boldsymbol{S}) \times \boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S}) \times \boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right) \times \boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$$

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \boldsymbol{P}(S_1^K|\boldsymbol{S})\ \boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$$

短语划分模型

短语翻译模型

短语调序模型

目标语言模型

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \underbrace{\boldsymbol{P}(S_1^K|\boldsymbol{S})}_{\downarrow} \underbrace{\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})}\underbrace{\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)}\underbrace{\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)}$$

短语划分模型

目标：将一个词序列如何划分为短语序列

方法：一般假设每一种短语划分方式都是等概率的

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \underline{\boldsymbol{P}(S_1^K|\boldsymbol{S})}\ \underline{\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})}\ \underline{\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)}\ \underline{\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)}$$

剩下的三个核心模型：

1. 短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})$
2. 短语调序模型： $\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)$
3. 目标语言模型： $\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$

基于短语的统计机器翻译

短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K, \boldsymbol{S})$

1. 如何学习短语翻译规则
2. 如何估计短语翻译概率

双语句对词语对齐

短语翻译规则抽取

(澳洲 是, Australia is)

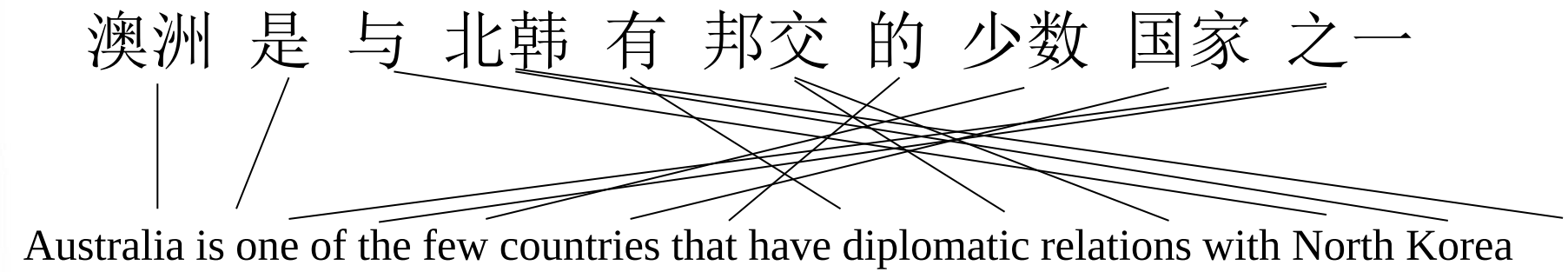
(与 北韩, with North Korea)

(有 邦交, have the diplomatic relations)

(的 少数 国家 之一, one of the few countries that)

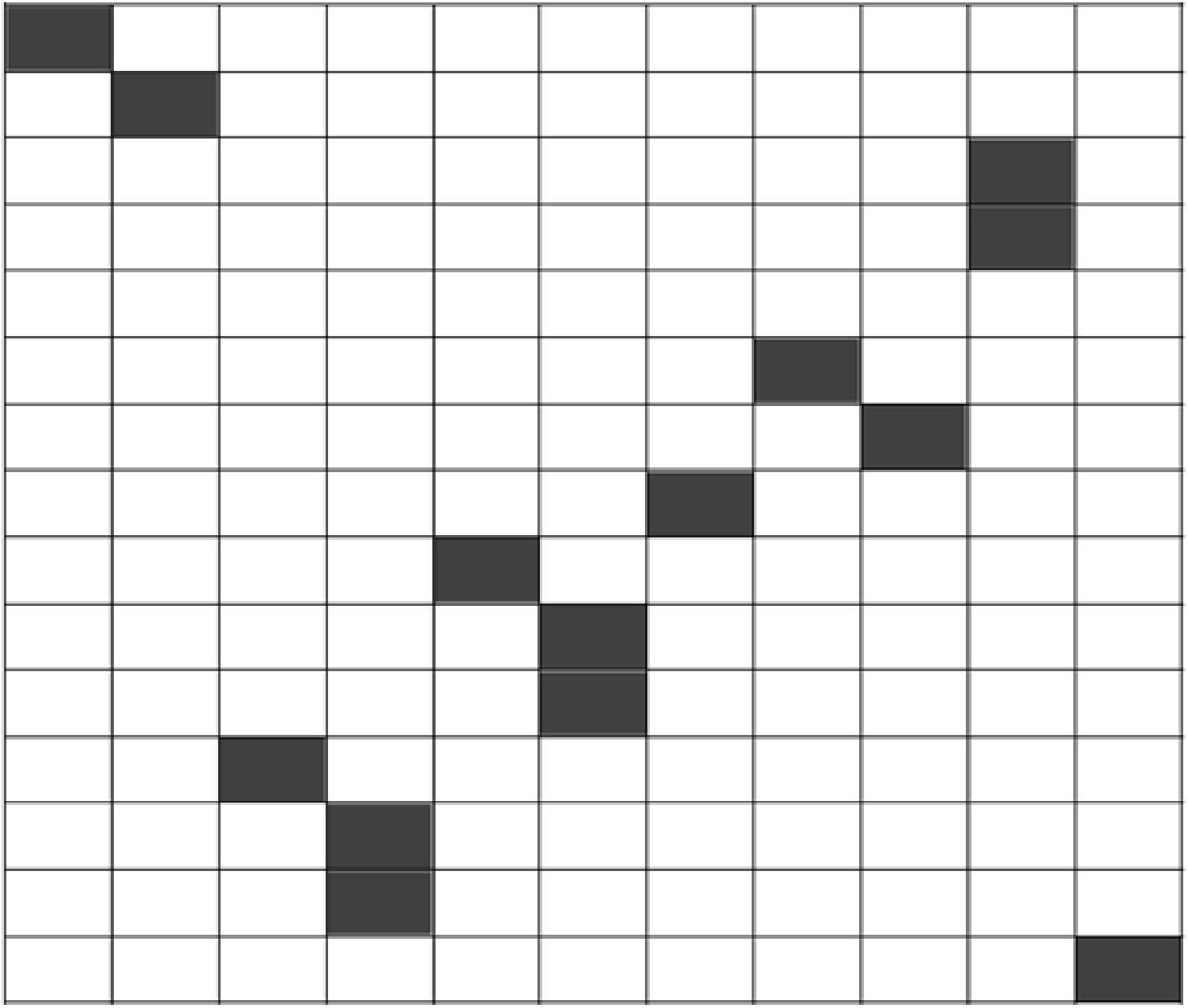
基于短语的统计机器翻译

双语句对词语对齐



IBM model 1-5

澳洲 是 与 北韩 有 邦交 的 少数 国家 之一 。



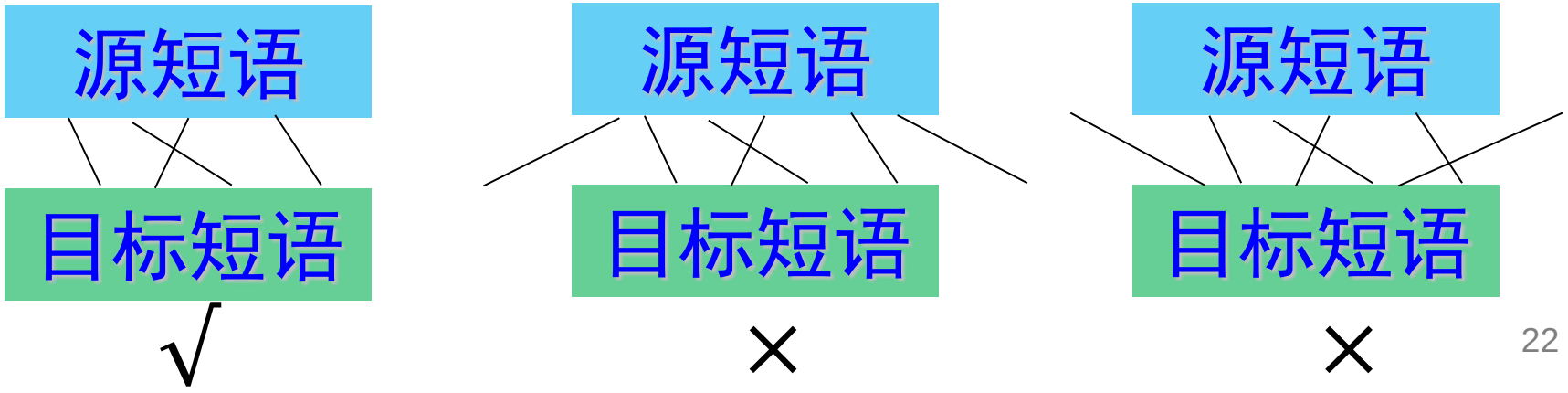
Australia
is
one
of
the
few
countries
that
have
diplomatic
relations
with
North
Korea
.

基于短语的统计机器翻译

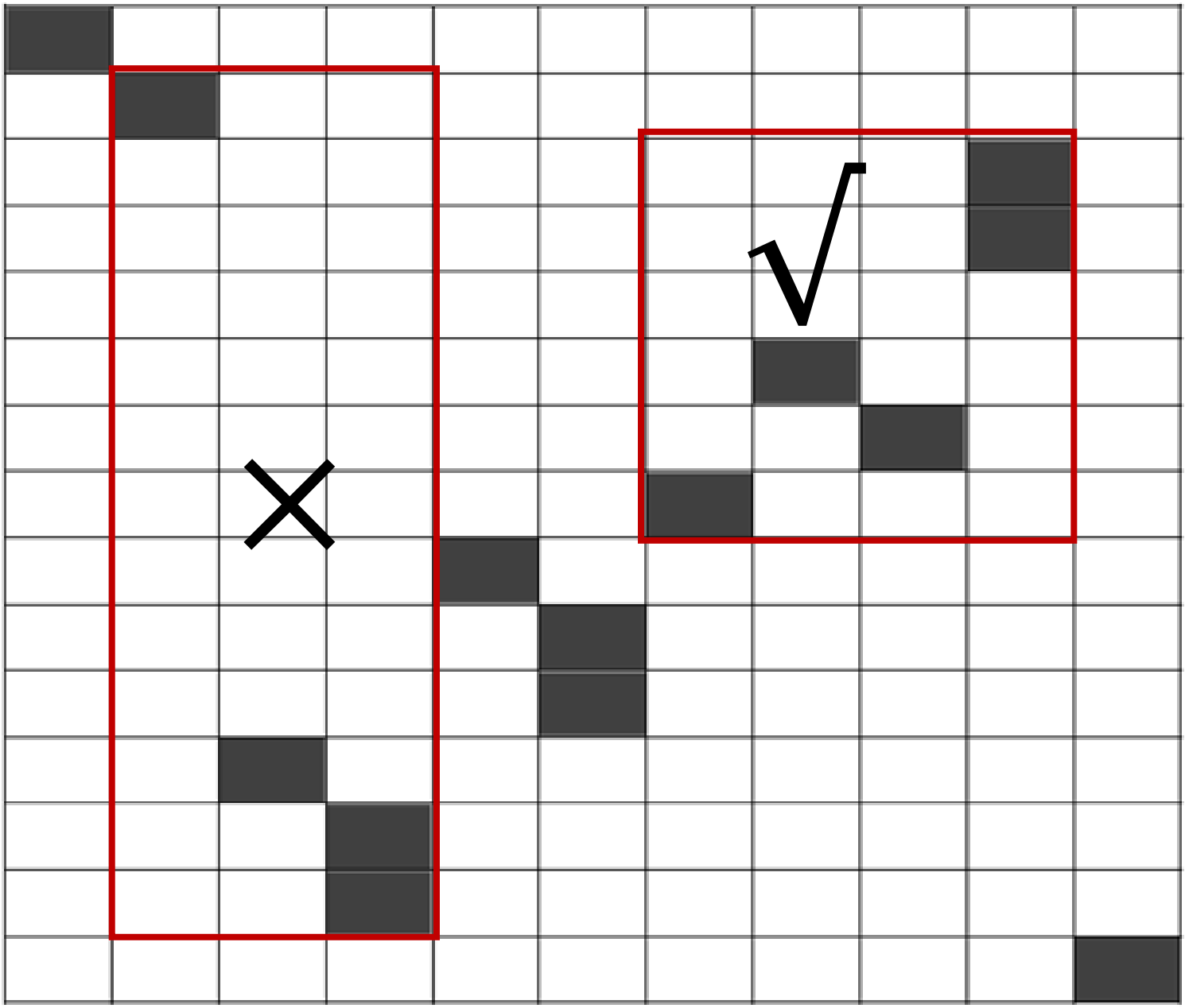
短语翻译规则抽取

算法：对于源语言句子S中的任一短语S_i^j，根据词语对齐A找到目标语言句子T中的对齐片段$T_{i'}^{j'}$，若S_i^j与$T_{i'}^{j'}$满足对齐一致性，则$\left(S_i^j, T_{i'}^{j'}\right)$为一条短语翻译规则。

对齐一致性：S_i^j中每个词S_k，若$(k,k') \in A$，则$i' \le k' \le j'$，$T_{i'}^{j'}$中每个词$T_{t'}$，若$(t,t') \in A$，则$i \le t \le j$。



澳洲 是 与 北韩 有 邦交 的 少数 国家 之一 。



Australia is one of the few countries that have diplomatic relations with North Korea .

基于短语的统计机器翻译

短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K, \boldsymbol{S})$

1. 如何学习短语翻译规则 — 双语句对词语对齐
2. 如何估计短语翻译概率 — 短语翻译规则抽取

(澳洲 是, Australia is)

(与 北韩, with North Korea)

(有 邦交, have the diplomatic relations)

(的 少数 国家 之一, one of the few countries that)

基于短语的统计机器翻译

短语翻译概率估计：4个翻译概率 （最大似然）

1. 正向、逆向短语翻译概率 $\boldsymbol{p(t|s), p(s|t)}$
2. 正向、逆向词汇化翻译概率 $\boldsymbol{p_{lex}(t|s), p_{lex}(s|t)}$

(与 北韩, with North Korea)

(与 北韩, and North Korea)

(和 北韩, with North Korea)

(与 朝鲜, with North Korea)

$$p(t|s) = \frac{1}{2}$$

$$p(s|t) = \frac{1}{3}$$

基于短语的统计机器翻译

短语翻译概率估计：4个翻译概率 （最大似然）

1. 正向、逆向短语翻译概率 $\boldsymbol{p(t|s), p(s|t)}$
2. 正向、逆向词汇化翻译概率 $\boldsymbol{p_{lex}(t|s), p_{lex}(s|t)}$

(与 北韩, with North Korea)

$$\boldsymbol{p(with|与) = 0.4}$$

$$\boldsymbol{p(North|北韩) = 0.1}$$

$$\boldsymbol{p(Korea|北韩) = 0.5}$$

$$\boldsymbol{p_{lex}(t|s)} = \prod_{j=1}^{|t|} \frac{1}{|\{i|(j,i) \in A\}|} \sum_{\forall (j,i) \in A} \boldsymbol{p(t_j|s_i)}$$

$$\boldsymbol{p_{lex}(t|s) = 0.4 \times 0.1 \times 0.5 = 0.02}$$

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \underline{\boldsymbol{P}(S_1^K|\boldsymbol{S})}\ \underline{\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})}\ \underline{\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)}\ \underline{\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)}$$

剩下的三个核心模型：

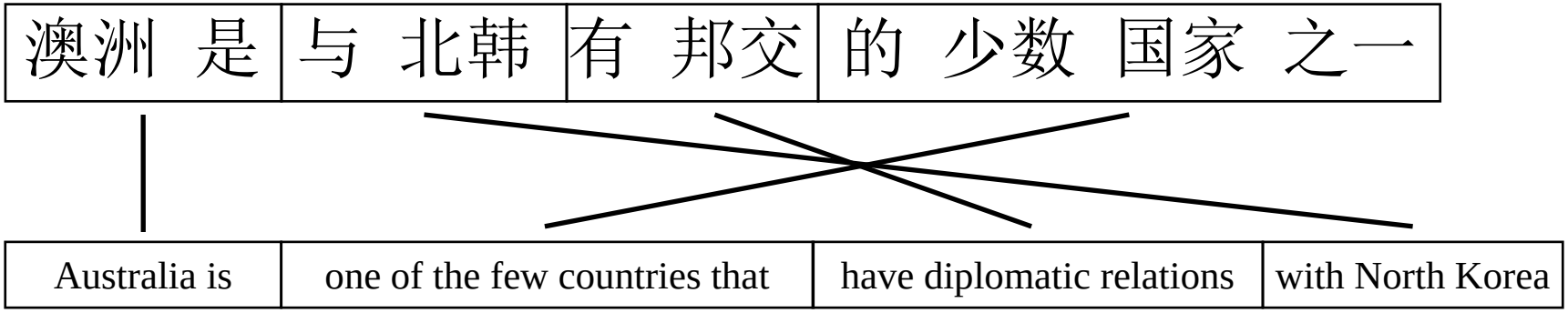
1. 短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})$
2. 短语调序模型： $\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)$
3. 目标语言模型： $\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$

基于短语的统计机器翻译

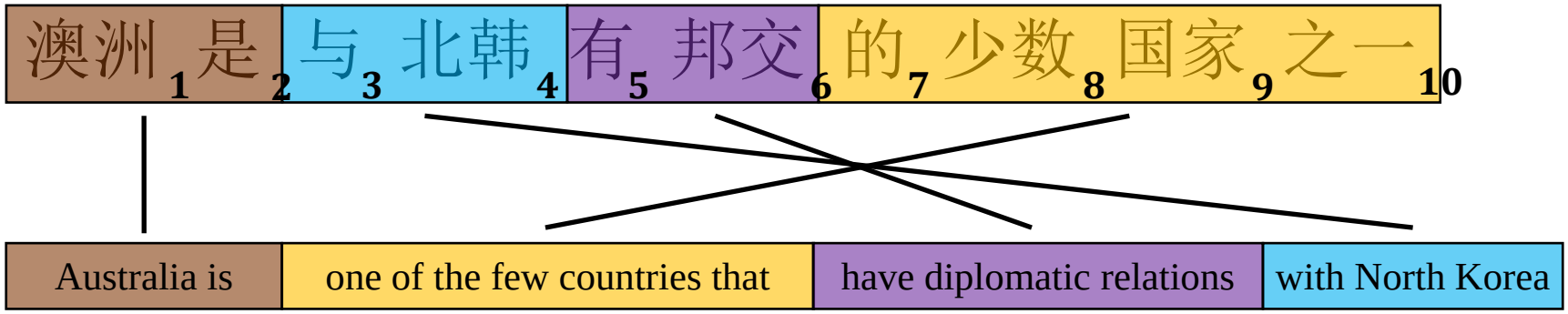
短语调序模型： $\left(T_1^{K'} | T_1^K, S_1^K, \boldsymbol{S}\right)$

两种常用方法：

1. 距离跳转模型
2. 分类模型



基于短语的统计机器翻译



Monotone：顺序拼接　　***Swap***：交换　　***Swap***：交换

(与 北韩, with North Korea)　(有 邦交, have the diplomatic relations)

↓ 特征提取

f0=与, f1=北韩, f2=有, f3=邦交, f4=with, f5=korea, f6=have, f7=relations

↓ 最大熵模型

Swap　　[Xiong et al., 2006]

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \underline{\boldsymbol{P}(S_1^K|\boldsymbol{S})}\ \underline{\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})}\ \underline{\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)}\ \underline{\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)}$$

剩下的三个核心模型：

1. 短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})$
2. 短语调序模型： $\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)$
3. 目标语言模型： $\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$

基于短语的统计机器翻译

$$P(w_1 w_2 \cdots w_{t-1} w_n)$$

$$= \prod_{t=1}^{n} P(w_t | w_{t-1} \cdots w_1)$$

$$\approx \prod_{t=1}^{n} P(w_t | w_{t-1} \cdots w_{t-n+1})$$

$$P(w_t | w_{t-1} \cdots w_{t-n+1})$$

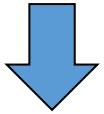
$$= \frac{count(w_{t-1} \cdots w_{t-n+1} w_t)}{count(w_{t-1} \cdots w_{t-n+1})}$$

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T, S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T, S_1^K, T_1^K, T_1^{K'}}{\operatorname{argmax}} \boldsymbol{P}(S_1^K|\boldsymbol{S})\boldsymbol{P}(T_1^K|S_1^K, \boldsymbol{S})\boldsymbol{P}\left(T_1^{K'}|T_1^K, S_1^K, \boldsymbol{S}\right)\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'}, T_1^K, S_1^K, \boldsymbol{S}\right)$$



$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S}) = \underset{T}{\operatorname{argmax}} \left\{ \sum_{\boldsymbol{1}}^{\boldsymbol{M}} \boldsymbol{\lambda_m h_m}(\boldsymbol{T}, \boldsymbol{S}) \right\}$$

基于短语的统计机器翻译

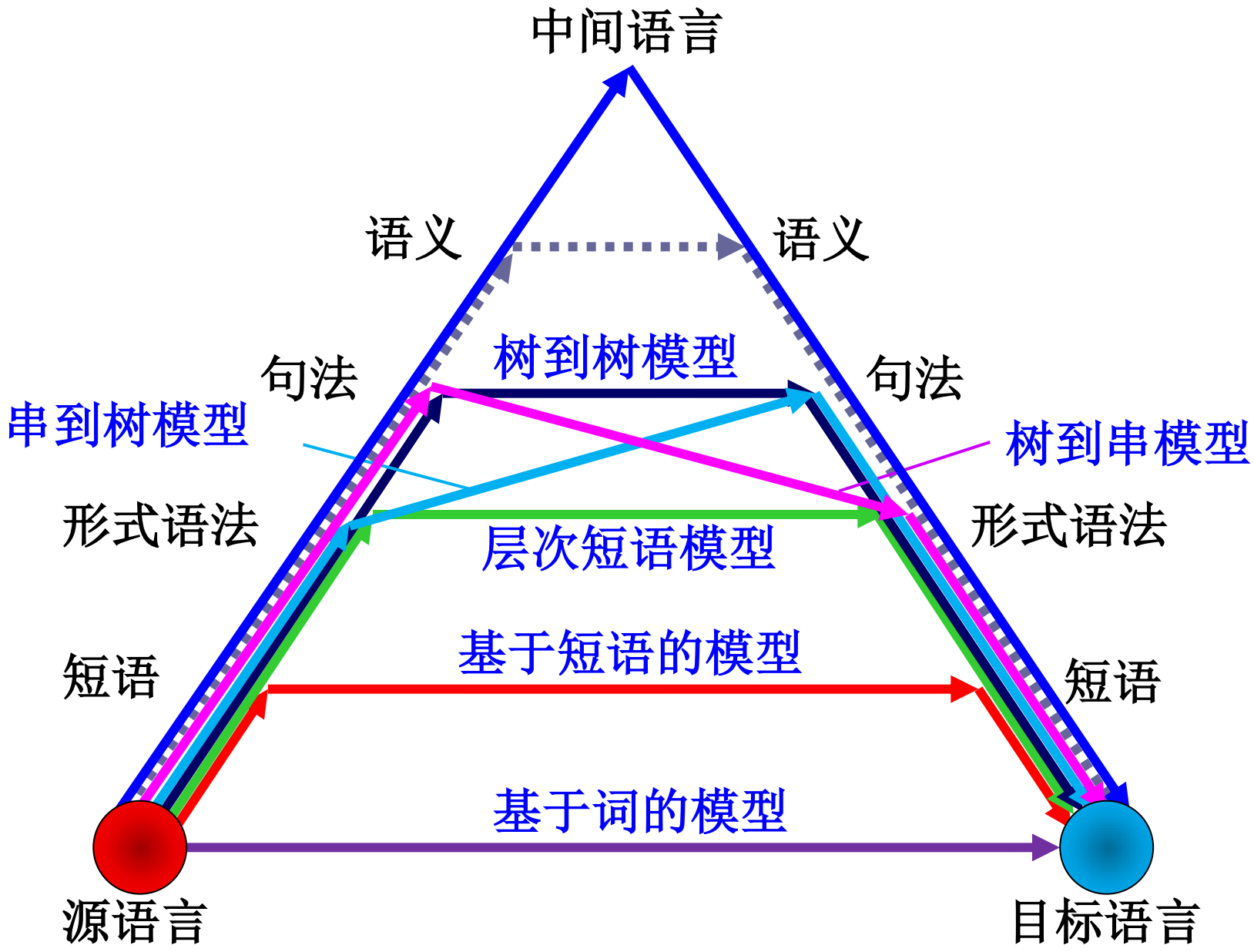
$$T' = \underset{T}{\operatorname{argmax}} P(T|S) = \underset{T}{\operatorname{argmax}} \left\{ \sum_{1}^{M} \lambda_m h_m(T,S) \right\}$$

$$h_1(T,S) = log\ p(t|s) \qquad h_5(T,S) = log\ P\left(T_1^{K'}|T_1^K, S_1^K, S\right)$$

$$h_2(T,S) = log\ p(s|t) \qquad h_6(T,S) = log\ P\left(T|T_1^{K'}, T_1^K, S_1^K, S\right)$$

$$h_3(T,S) = log\ p_{lex}(t|s) \qquad h_7(T,S) = log\ len(T)$$

$$h_4(T,S) = log\ p_{lex}(s|t) \qquad h_8(T,S) = log\ count(phrases) = log\ K$$


中间语言
语义
语义
句法
句法
树到树模型
串到树模型
树到串模型
形式语法
形式语法
层次短语模型
短语
短语
基于短语的模型
基于词的模型
源语言
目标语言

统计机器翻译中深度学习的应用

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \underline{\boldsymbol{P}(S_1^K|\boldsymbol{S})}\ \underline{\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})}\ \underline{\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)}\ \underline{\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)}$$

剩下的三个核心模型：

1. 短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})$
2. 短语调序模型： $\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)$
3. 目标语言模型： $\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$

基于计数的N-元语言模型

该 课程 很 枯燥， 大家 觉得 很 无聊 。

$$P(w_t|w_{t-1}\cdots w_{t-n+1}) \quad P(\text{无聊}|\text{很}) = \frac{count(\text{很 无聊})}{count(\text{很})}$$

$P(\text{无聊}|\text{很})$

vs.

$P(\text{枯燥}|\text{很})$

问题①：数据稀疏

N-元组“很 无聊”未出现过，则回退

问题②：忽略语义相似性

“无聊”与“枯燥”虽语义相似，但无法共享信息

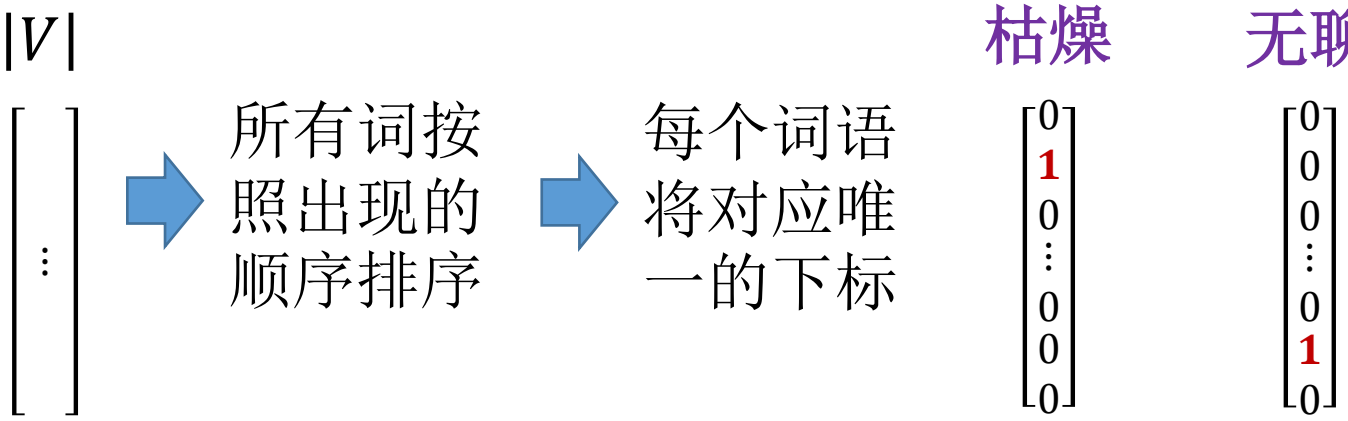
基于计数的N-元语言模型

- 典型方法：抽象符号（字符串）

  该 课程 很 枯燥， 大家 觉得 很 无聊 。

  w_0=该 w_1=课程 w_2=很 w_3=枯燥 w_4=，
  w_5=大家 w_6=觉得 w_7=很 w_8=无聊 w_9=。

- 等价表示方法：one-hot表示法



词语表示

- 问题

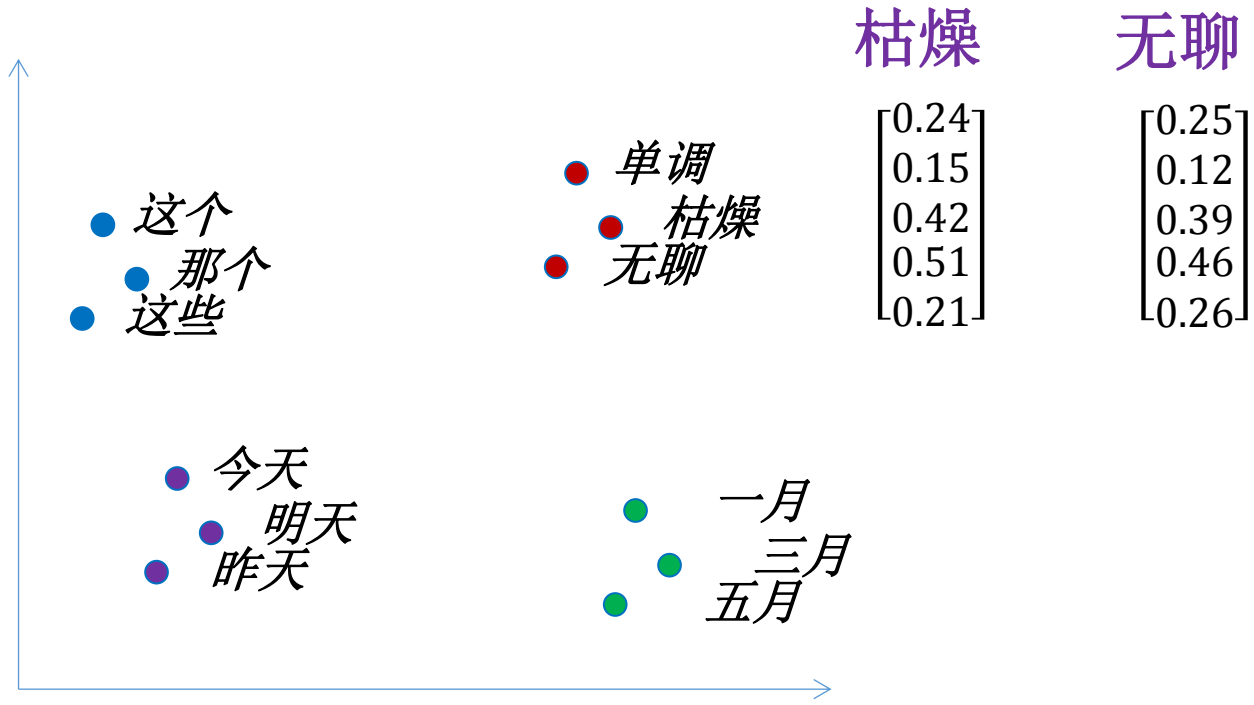
枯燥 $\begin{bmatrix}0\\1\\0\\\vdots\\0\\0\\0\end{bmatrix}$ 无聊 $\begin{bmatrix}0\\0\\0\\\vdots\\0\\1\\0\end{bmatrix}$

枯燥 $\otimes$ 无聊

$$\begin{bmatrix}0\\1\\0\\\vdots\\0\\0\\0\end{bmatrix} \times \begin{bmatrix}0\\0\\0\\\vdots\\0\\1\\0\end{bmatrix} = 0$$

任意两个词之间的相似度都为0！

词语表示



低维、稠密的连续实数空间

神经网络语言模型

很

$$\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\end{bmatrix}$$

P(无聊|很) ***vs.*** P(枯燥|很)

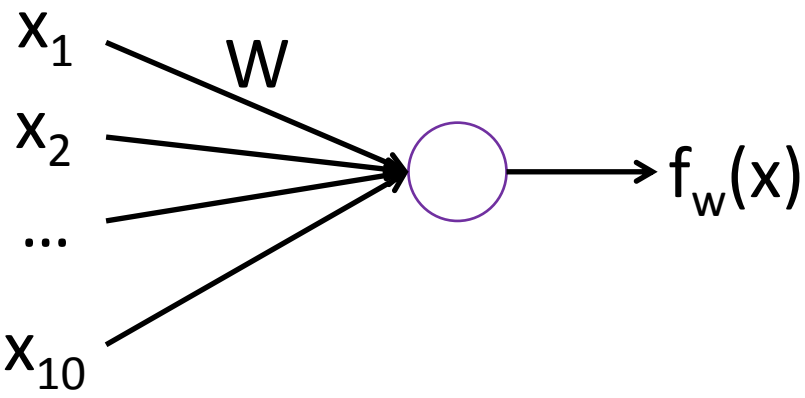
$$P\left(\begin{bmatrix}0.25\\0.12\\0.39\\0.46\\0.26\end{bmatrix}\middle|\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\end{bmatrix}\right) \quad \boldsymbol{vs.} \quad P\left(\begin{bmatrix}0.24\\0.15\\0.42\\0.51\\0.21\end{bmatrix}\middle|\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\end{bmatrix}\right)$$

$$f\left(\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\\0.25\\0.12\\0.39\\0.46\\0.26\end{bmatrix}\right) \quad \boldsymbol{vs.} \quad f\left(\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\\0.24\\0.15\\0.42\\0.51\\0.21\end{bmatrix}\right)$$

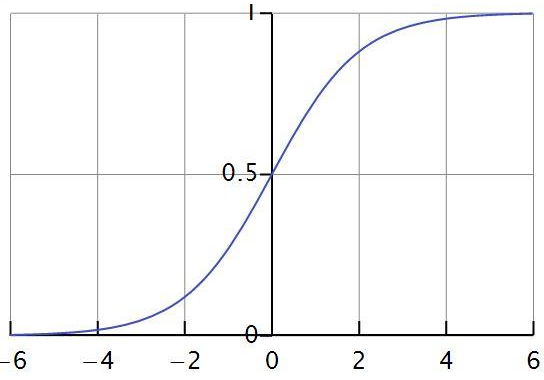
神经网络语言模型

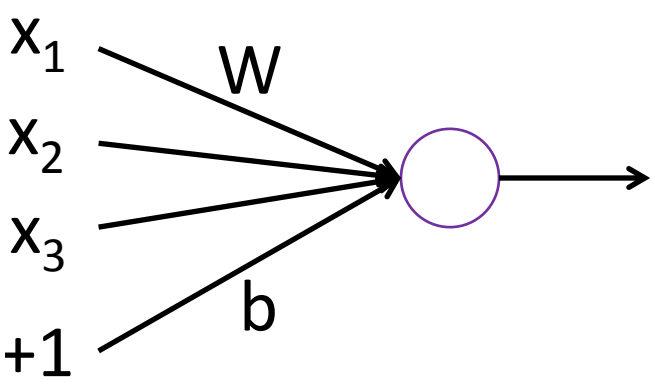
$$P(\text{无聊}|\text{很})$$

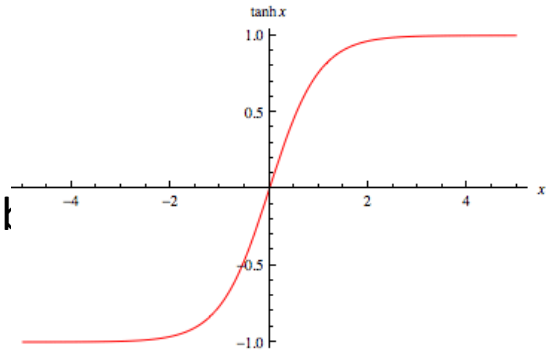
$$f\left(\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\\0.25\\0.12\\0.39\\0.46\\0.26\end{bmatrix}\right) = f\left(\begin{bmatrix}w_1 \times 0.01\\w_2 \times 0.59\\w_3 \times 0.18\\w_4 \times 0.05\\w_5 \times 0.47\\w_6 \times 0.25\\w_7 \times 0.12\\w_8 \times 0.39\\w_9 \times 0.46\\w_{10} \times 0.26\end{bmatrix}\right) = f(WX)$$



神经网络语言模型







$$h_{W,b}(x) = f(W^T x + b)$$

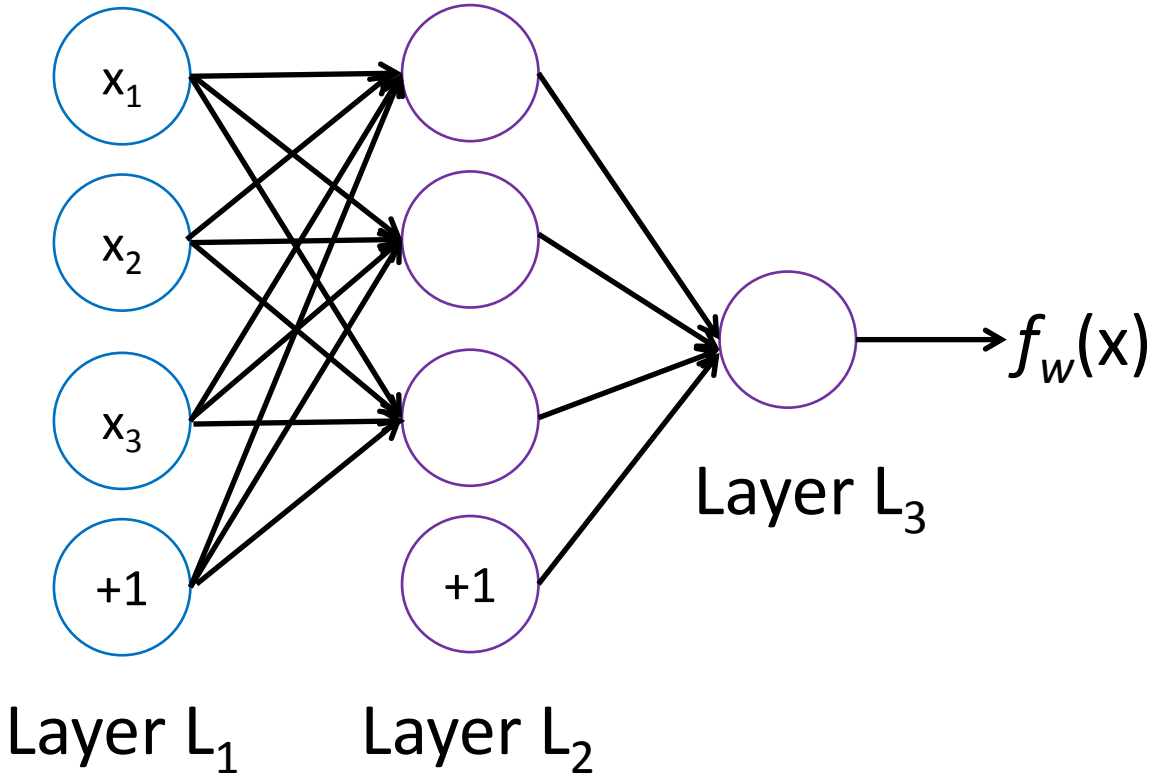
f: 非线性激活函数

$$\begin{cases} f(z) = \dfrac{1}{1 + exp(-z)} \\ f(z) = tanh(z) = \dfrac{e^z - e^{-z}}{e^z + e^{-z}} \end{cases}$$

$$f'(z) = f(z)\big(1 - f(z)\big)$$

$$f'(z) = 1 - f^2(z)$$

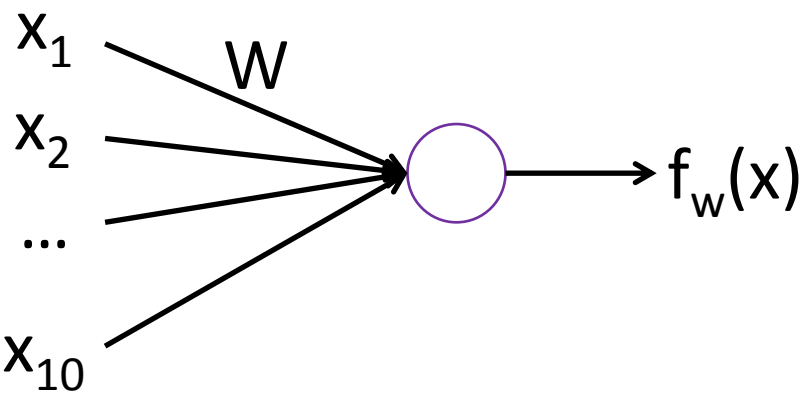
神经网络语言模型



神经网络语言模型

$P(\text{无聊}|\text{很})$ ***vs.*** $P(\text{枯燥}|\text{很})$

$$f\left(\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\\0.25\\0.12\\0.39\\0.46\\0.26\end{bmatrix}\right) = f\left(\begin{bmatrix}w_1 \times 0.01\\w_2 \times 0.59\\w_3 \times 0.18\\w_4 \times 0.05\\w_5 \times 0.47\\w_6 \times 0.25\\w_7 \times 0.12\\w_8 \times 0.39\\w_9 \times 0.46\\w_{10} \times 0.26\end{bmatrix}\right) = f(WX)$$



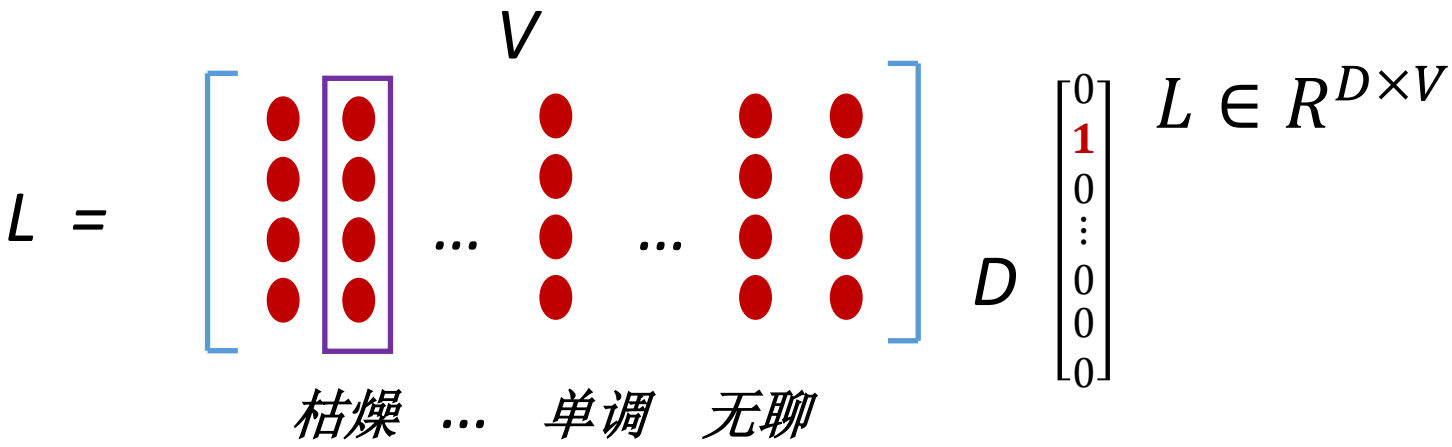
问题①：词向量

如何将每个词映射到实数向量空间中的一个点

问题②：*f*函数的设计

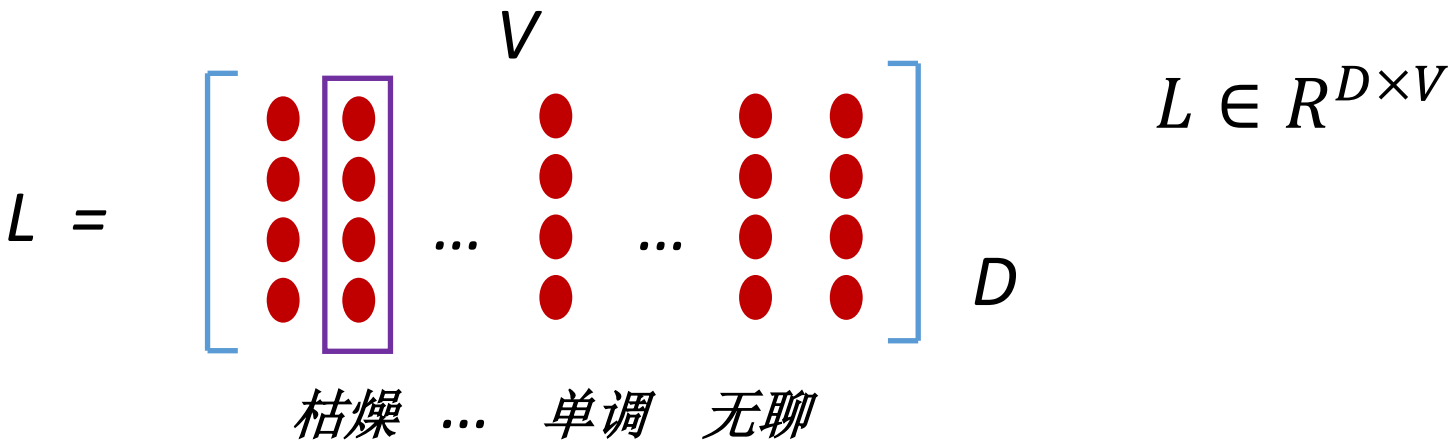
设计什么样的神经网络结构模拟函数*f*

神经网络语言模型-词向量



- 通常称为look-up table
  - 我们可以对L右乘一个词的one-hot表示 e 得到该词的低维、稠密的实数向量表达：$x = Le$

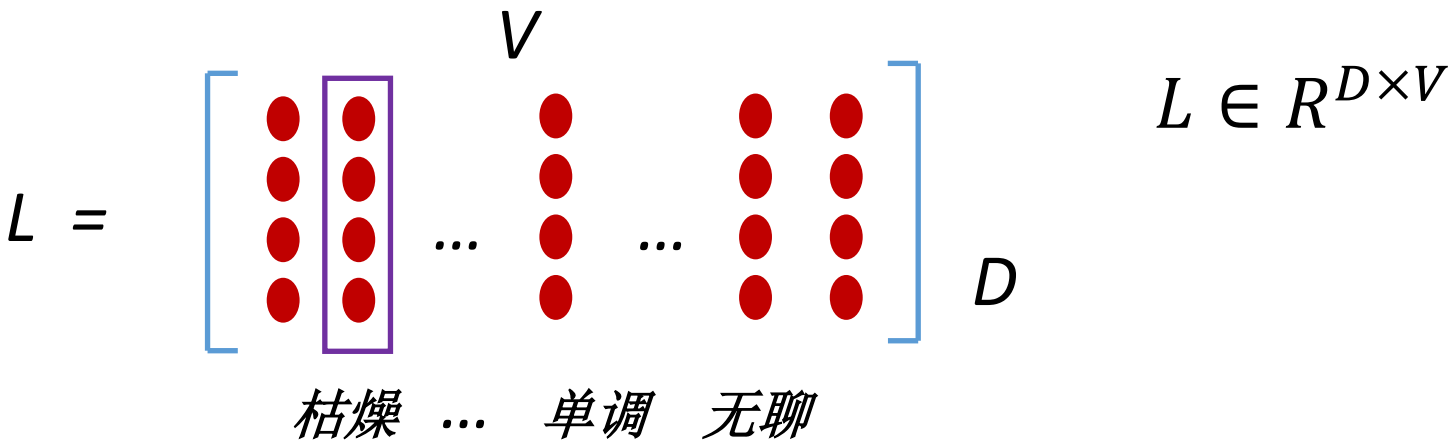
神经网络语言模型-词向量



$L \in R^{D \times V}$

- 词表规模V和词向量维度D如何确定
  - V的确定：1,训练数据中所有词；2, 频率高于某个阈值的所有词；3, 前V个频率最高的词
  - D的确定：超参数，人工设定，一般从几十到几百

神经网络语言模型-词向量



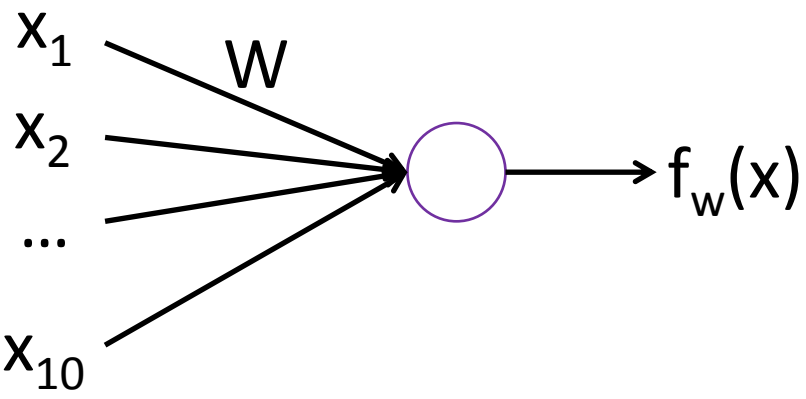
$L \in R^{D \times V}$

- 如何学习L
  - 通常先随机初始化，然后通过目标函数优化词的向量表达（e.g. 最大化语言模型似然度）

神经网络语言模型

$P(\text{无聊}|\text{很})$ ***vs.*** $P(\text{枯燥}|\text{很})$

$$f\left(\begin{bmatrix}0.01\\0.59\\0.18\\0.05\\0.47\\0.25\\0.12\\0.39\\0.46\\0.26\end{bmatrix}\right) = f\left(\begin{bmatrix}w_1 \times 0.01\\w_2 \times 0.59\\w_3 \times 0.18\\w_4 \times 0.05\\w_5 \times 0.47\\w_6 \times 0.25\\w_7 \times 0.12\\w_8 \times 0.39\\w_9 \times 0.46\\w_{10} \times 0.26\end{bmatrix}\right) = f(WX)$$



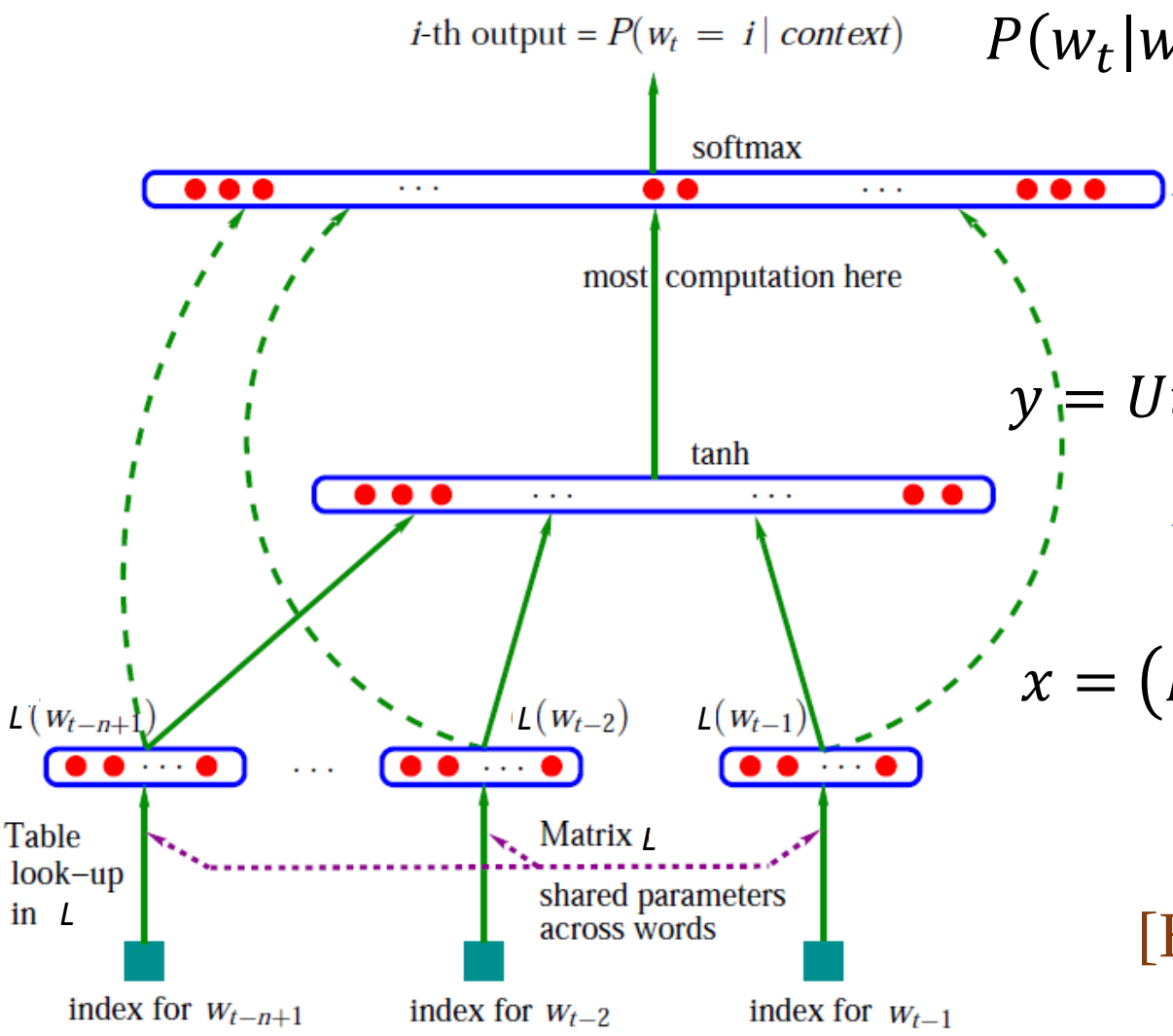
问题①：词向量

如何将每个词映射到实数向量空间中的一个点

问题②：*f*函数的设计

设计什么样的神经网络结构模拟函数*f*

前馈神经网络语言模型



$$P(w_t|w_{t-1}\cdots w_{t-n+1}) = \frac{e^{y_{w_t}}}{\sum_i e^{y_i}}$$

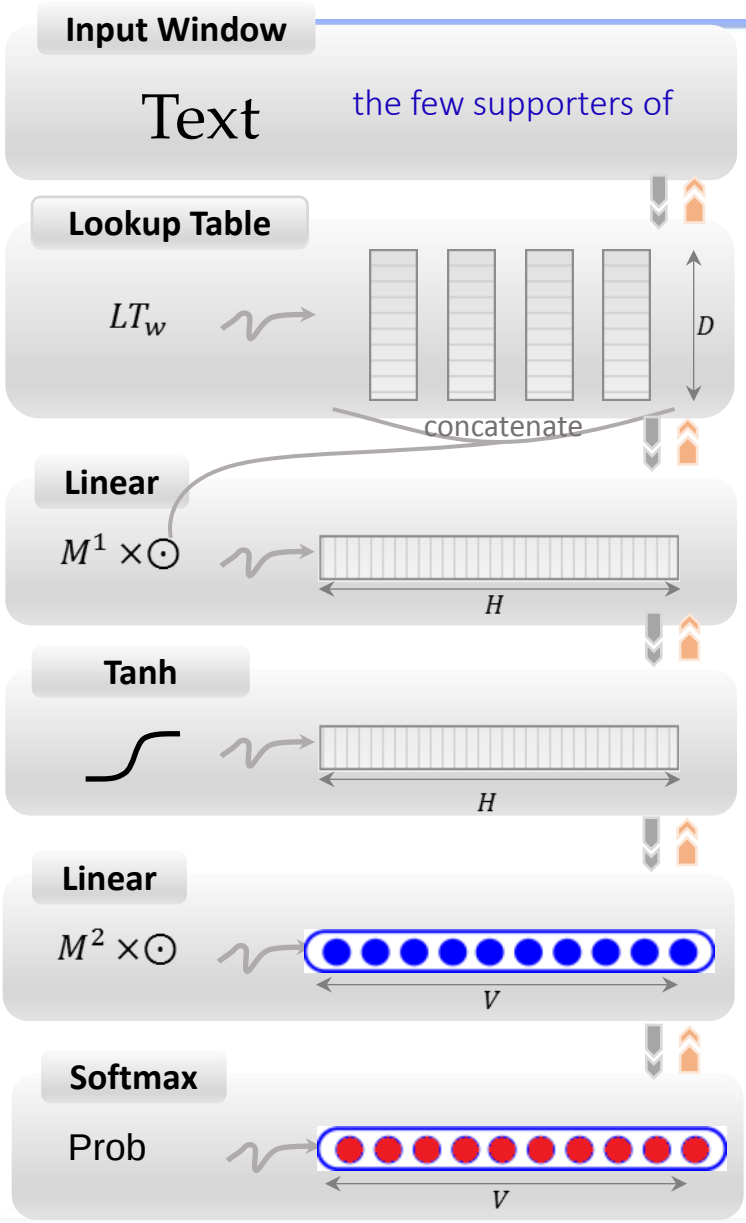
$$\theta \leftarrow \theta + \frac{\partial logP}{\partial \theta}$$

$$y = Utanh(Hx + d) + Wx + b$$

$$x = \left(L(w_{t-1}), \cdots L(w_{t-n+1})\right)$$

[Bengio et al., 2003]

前馈神经网络语言模型



$P(this|the, few, supporters, of)$

将每个词通过词向量矩阵L映射为低维实数向量

of → *(0.23, 0.15, 0.08, 0.31, .., 0.42)*

拼接所有词的向量，形成一个向量

隐藏层:
线性映射+非线性变换

⋮

***Softmax* 输出层:**

$P(this|the, few, supporters, of)$

前馈神经网络语言模型

- 问题

仅对小窗口的历史信息建模

例如5-gram语言模型，仅考虑前面4个词的历史信息

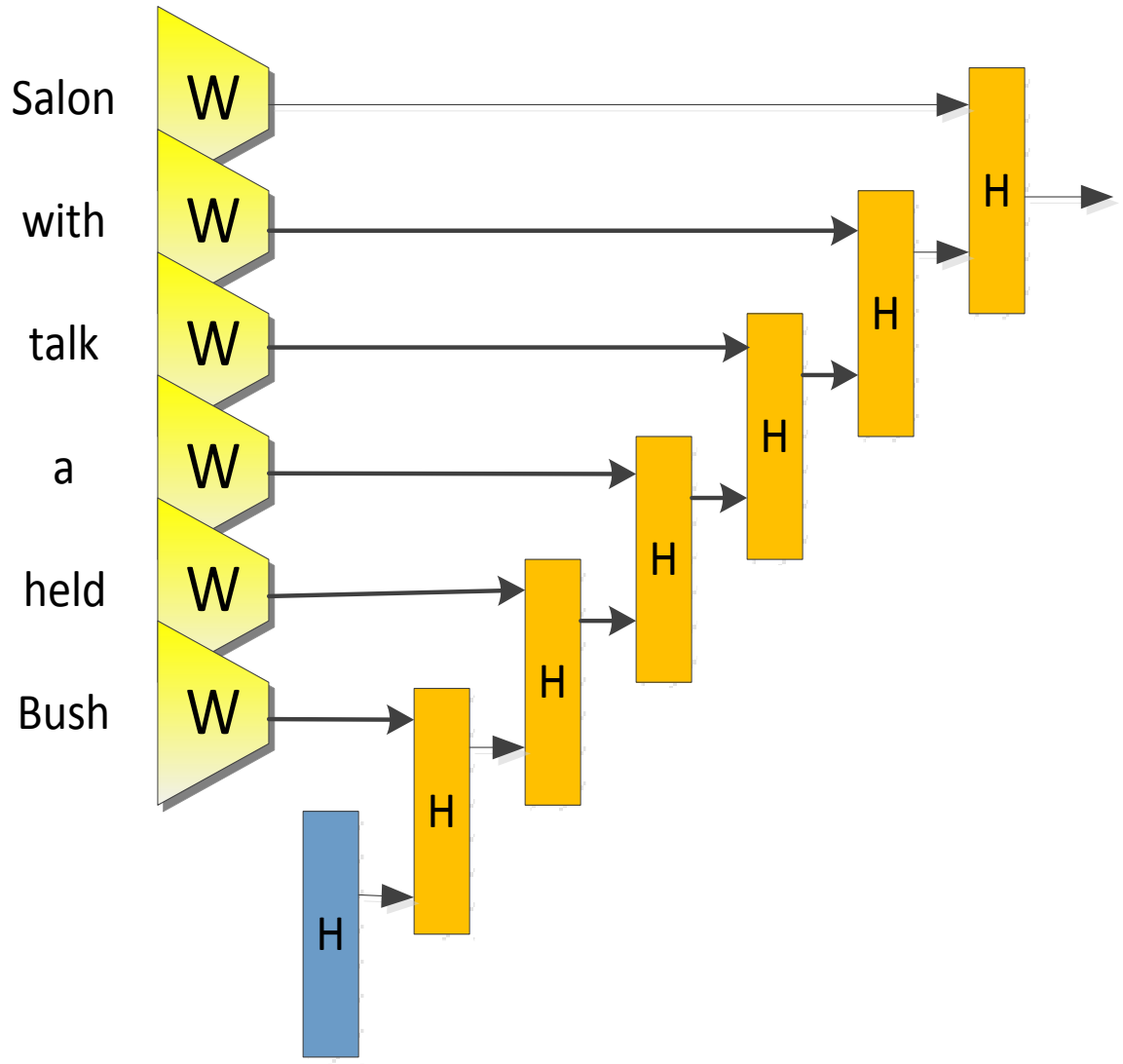
$$P(w_t|w_{t-1}\cdots w_{t-n+1})$$

能否对所有的历史信息进行建模

即第t个词的语言模型概率依赖于所有前t-1个词

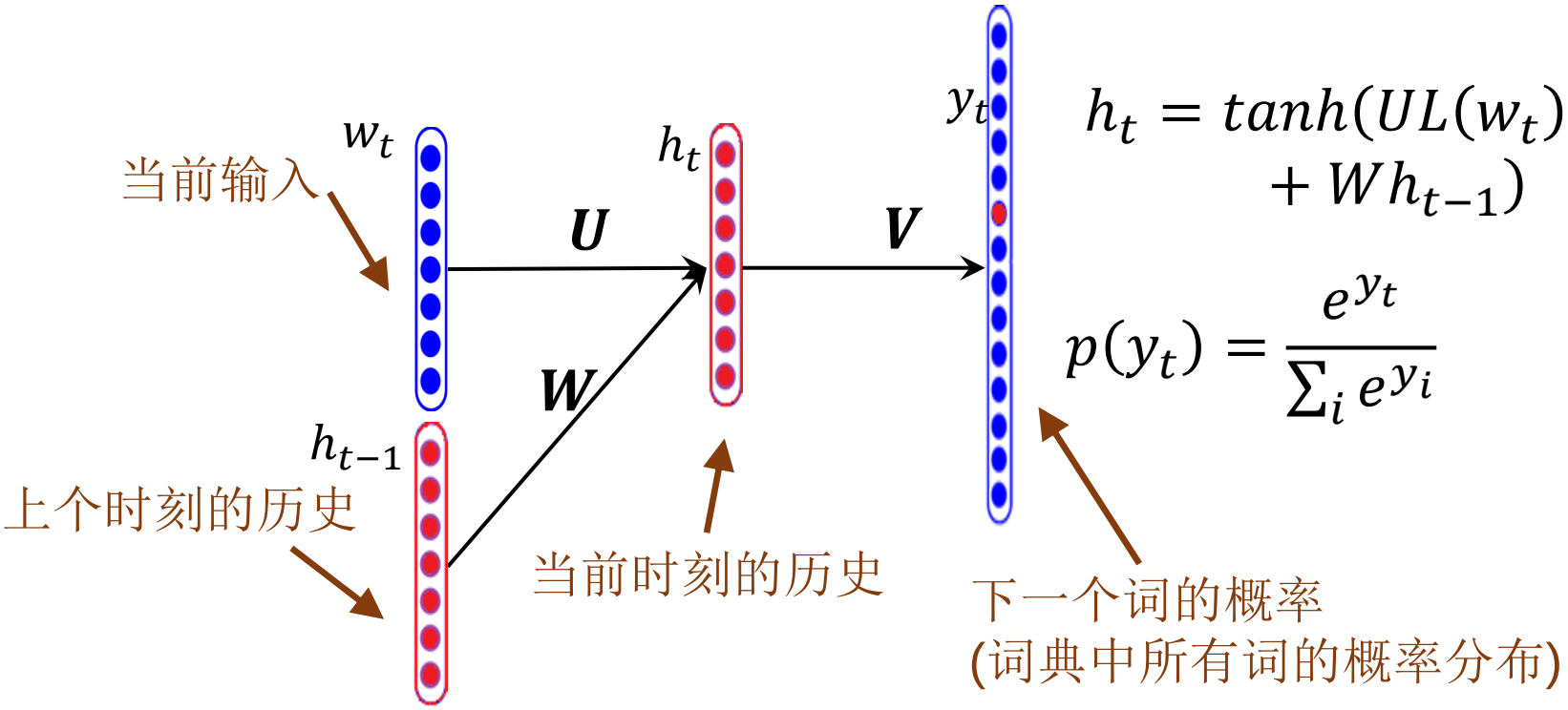
$$P(w_t|w_{t-1}\cdots w_2 w_1)$$

循环神经网络语言模型

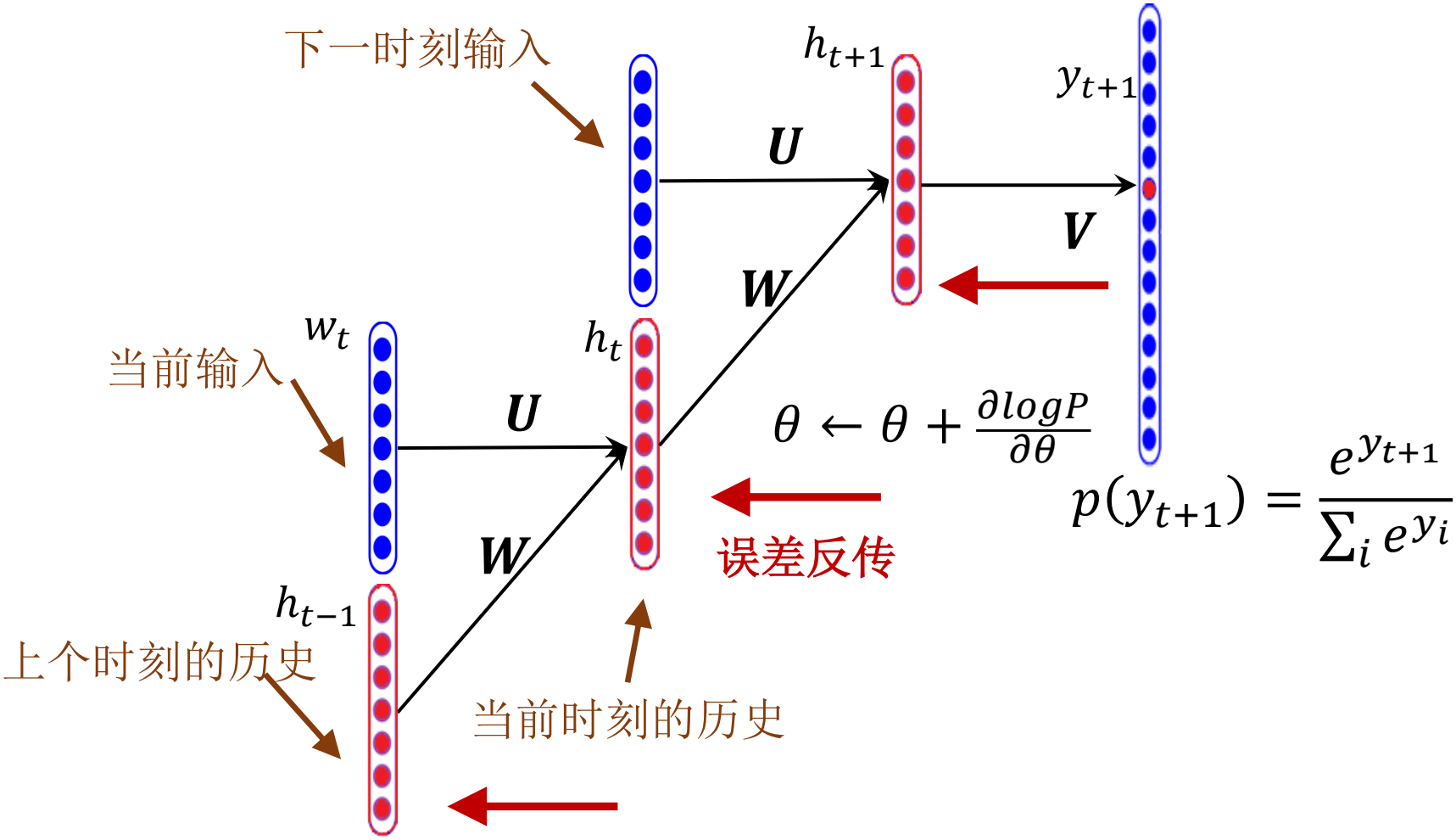


循环神经网络语言模型

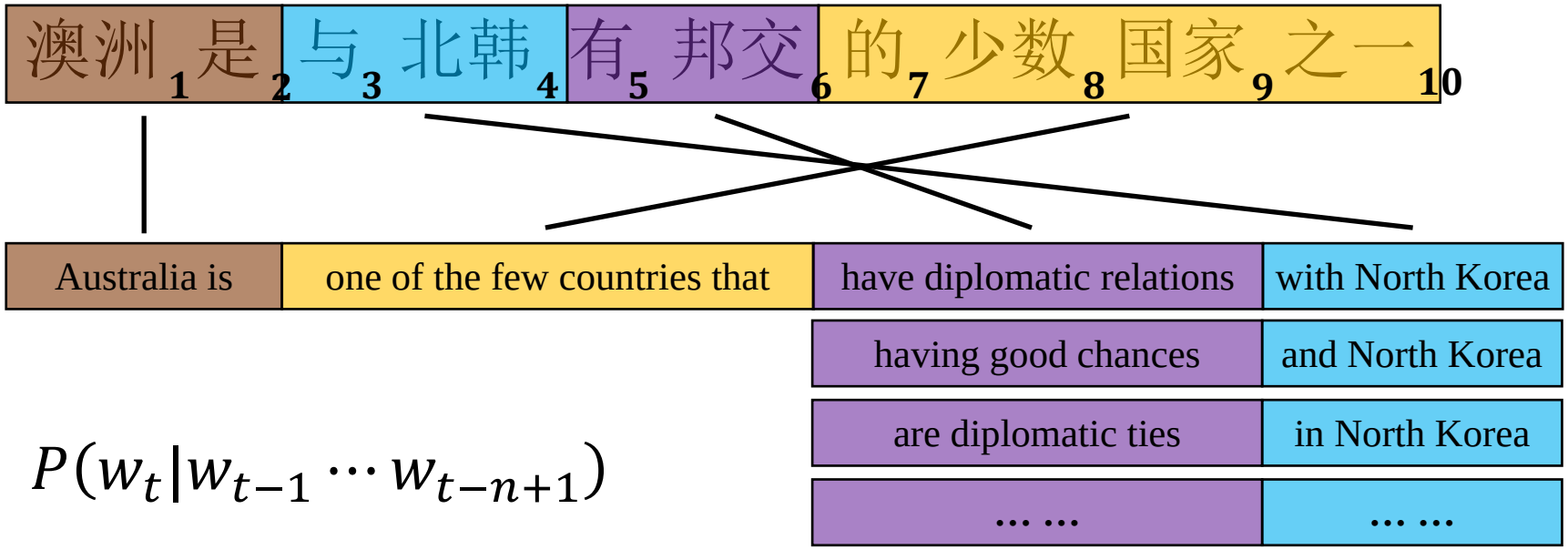
- 输入: $t-1$时刻历史 h_{t-1} 与 t时刻输入 w_t
- 输出: t时刻历史 h_t 与 下个时刻$t+1$输入y_t 的概率



循环神经网络语言模型


下一时刻输入
h_{t+1}
y_{t+1}
U
V
W
w_t
h_t
当前输入
U
$\theta \leftarrow \theta + \frac{\partial logP}{\partial \theta}$
$p(y_{t+1}) = \frac{e^{y_{t+1}}}{\sum_i e^{y_i}}$
误差反传
W
h_{t-1}
上个时刻的历史
当前时刻的历史

神经网络语言模型-融入解码器



$$P(w_t|w_{t-1}\cdots w_{t-n+1})$$

解码中计算语言模型概率：①翻译候选多；②自底往上的解码方式无法知道译文前缀。

- 适合前馈神经网络

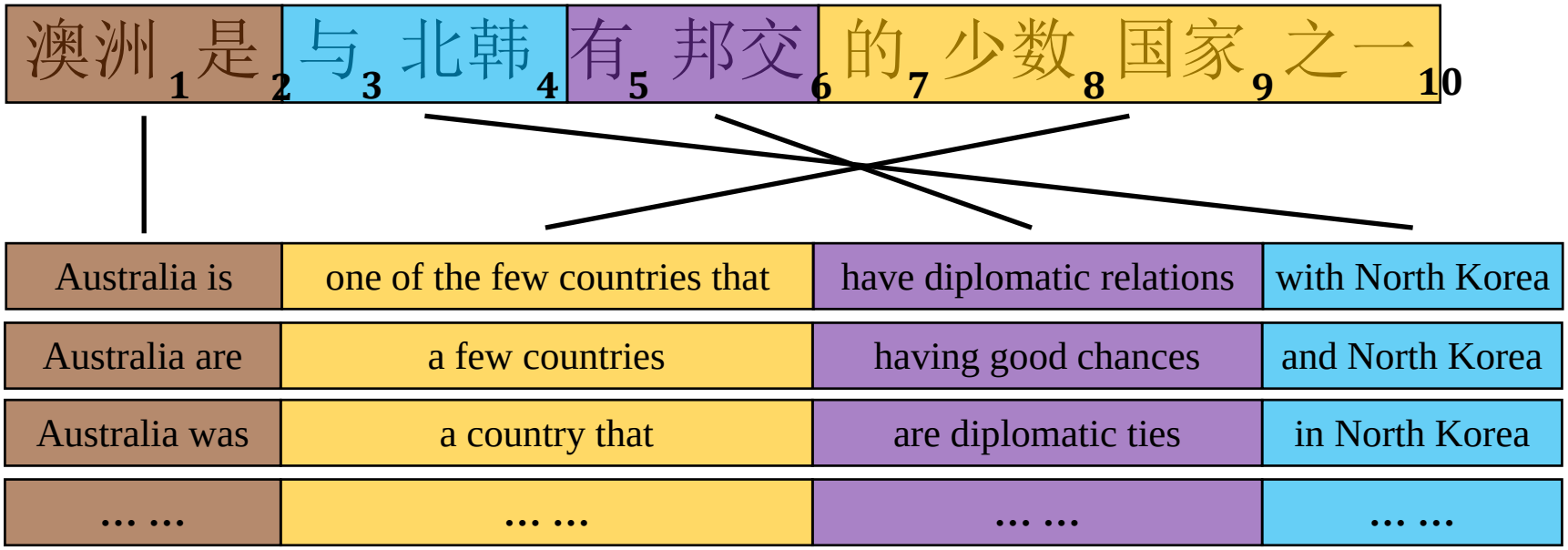
神经网络语言模型-融入解码器

- 汉语-英语：NIST-2012 受限评测

Setting	dev	2004	2005	2006
baseline	38.2	38.4	37.7	34.3
+NNLM decoding	**39.1**	**39.5**	**38.8**	**34.9**

[Vaswani et al., EMNLP-2013]

神经网络语言模型-译文重排序



对多个翻译结果重排序：①翻译候选少；②可以利用译文前缀信息。

- 适合循环神经网络

神经网络语言模型-译文重排序

- 法语-英语：WMT-2012

Setting	dev	news2010	news2011	com2011	Ave.
baseline	26.6	27.6	28.3	27.5	27.8
+RNNLM (2m)	**27.5**	**28.1**	**28.6**	**28.1**	**28.3**
+RNNLM (50m)	**27.7**	**28.2**	**29.0**	**28.1**	**28.5**

[Auli et al., EMNLP-2013]

神经网络语言模型-译文重排序

- 德语-英语：WMT-2012

Setting	dev	news2010	news2011	com2011	Ave.
baseline	21.2	20.7	19.2	20.6	20.0
+RNNLM (2m)	**21.8**	**20.9**	**19.4**	**20.9**	**20.3**
+RNNLM (50m)	**22.1**	**21.1**	**19.7**	**21.0**	**20.5**

[Auli et al., EMNLP-2013]

神经网络语言模型-译文重排序

- 英语-德语：WMT-2012

Setting	dev	news2010	news2011	com2011	Ave.
baseline	15.2	15.6	14.3	15.7	15.1
+RNNLM (2m)	**15.7**	**15.9**	**14.6**	**16.0**	**15.4**
+RNNLM (50m)	**15.8**	**15.9**	**14.7**	**16.1**	**15.5**

[Auli et al., EMNLP-2013]

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \underline{\boldsymbol{P}(S_1^K|\boldsymbol{S})}\ \underline{\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})}\ \underline{\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)}\ \underline{\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)}$$

剩下的三个核心模型：

1. 短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})$
2. 短语调序模型： $\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)$
3. 目标语言模型： $\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$

基于最大似然的短语翻译概率估计

短语翻译概率估计：4个翻译概率 （最大似然）

1. 正向、逆向短语翻译概率 $\boldsymbol{p(t|s), p(s|t)}$
2. 正向、逆向词汇化翻译概率 $\boldsymbol{p_{lex}(t|s), p_{lex}(s|t)}$

(与 北韩, with North Korea)

100　　1

$$p(t|s) = \frac{1}{100}$$

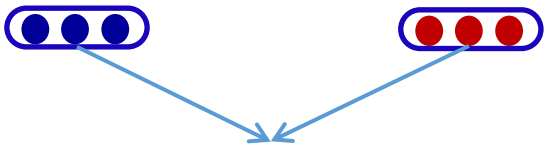
最大似然概率估计无法刻画双语短语之间的语义相似程度！

基于语义向量空间的翻译置信度估计

短语翻译概率估计：4个翻译概率 （最大似然）

1. 正向、逆向短语翻译概率 $\boldsymbol{p(t|s), p(s|t)}$
2. 正向、逆向词汇化翻译概率 $\boldsymbol{p_{lex}(t|s), p_{lex}(s|t)}$

(与 北韩, with North Korea)

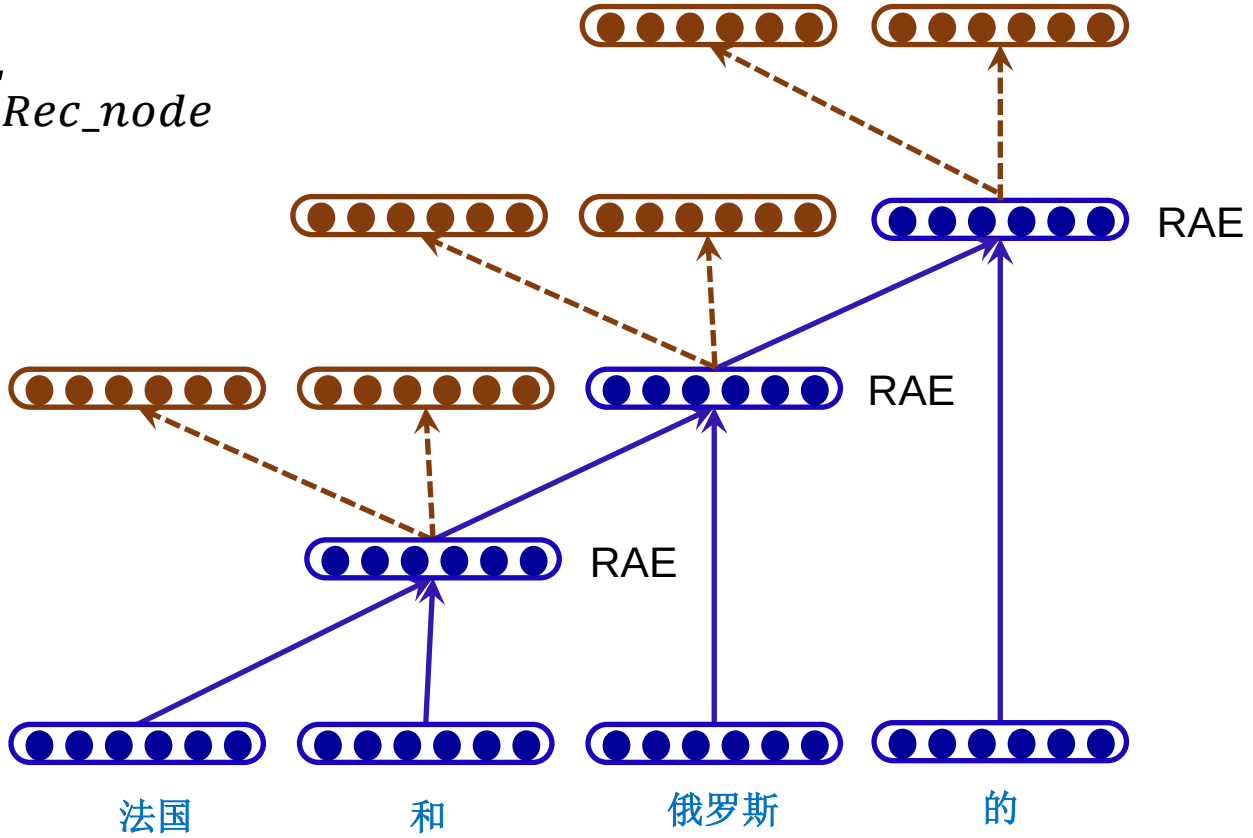


可以在语义向量空间中计算双语短语的语义相似度

基于语义向量空间的短语表示

- 递归自动编码器
- 目标函数：最小化所有节点的重构误差

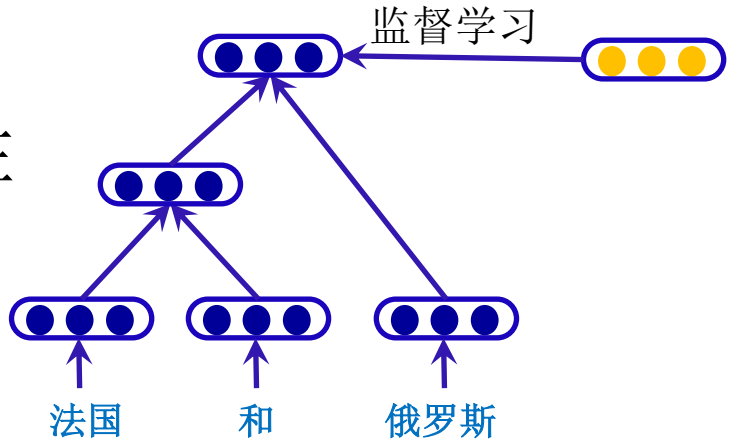
$$E_{total} = \sum_{node} E_{Rec\_node}$$



基于语义向量空间的短语表示

- 理想方法：有标注数据

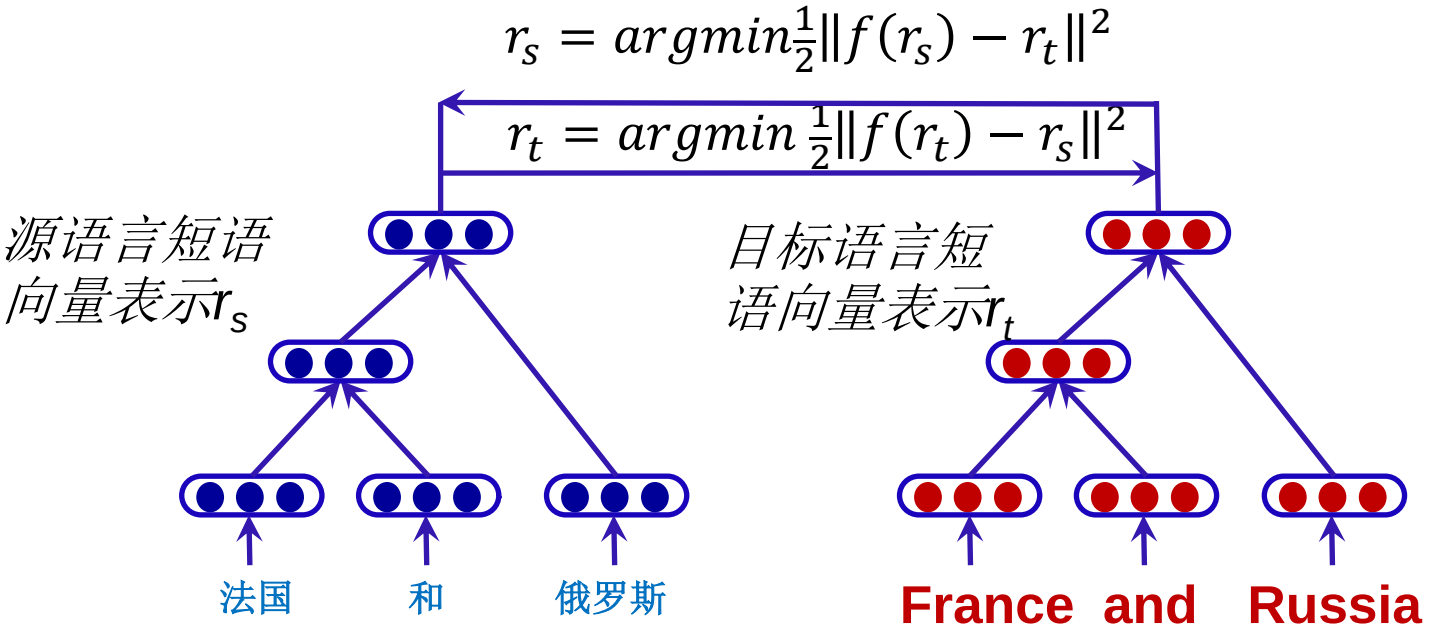
- 但是， 现实中不存在正确标注的短语向量



基于语义向量空间的短语表示

- 假设
  - 短语与其翻译具有相同的语义向量表示
- 目标函数
  - 最小化短语翻译对间的语义表示误差
- 模型
  - Pre-training：无监督递归自编码器学习短语初始表示
  - Fine-tuning：相互监督学习，优化短语向量表示

基于语义向量空间的短语表示



基于语义向量空间的短语表示

- 目标函数

正则化项

$$J = E(S,T;\theta) + \frac{1}{2}\|\lambda\|^2$$

重构误差

双语语义误差

$$E(S,T;\theta) = \alpha E_{rec}(S,T;\theta) + (1-\alpha)E_{regression}(S,T;\theta)$$

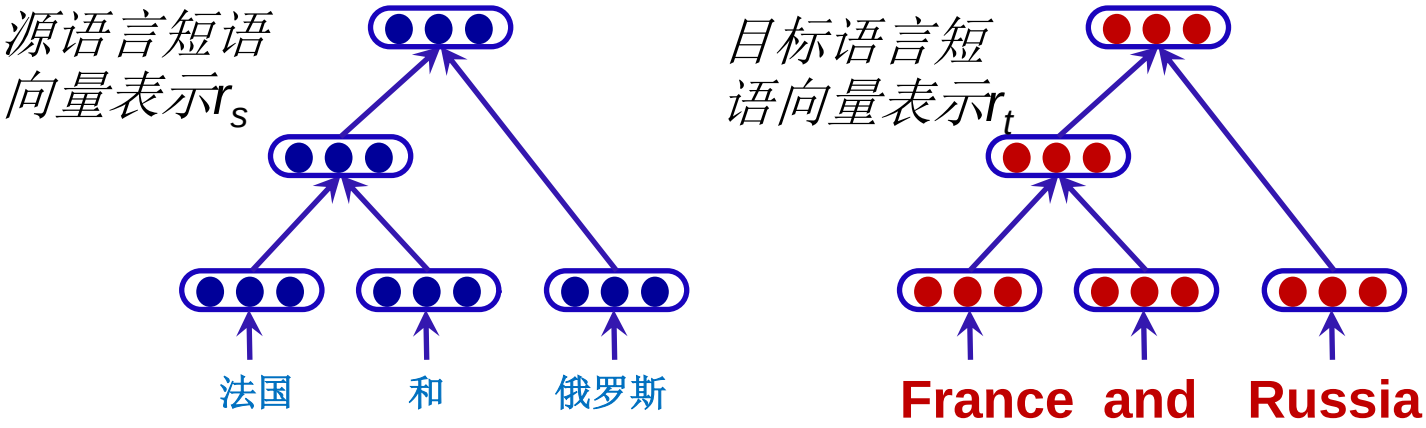
$$E_{rec}(S,T;\theta) = E_{rec}(S;\theta) + E_{rec}(T;\theta)$$

$$E_{regression}(S,T;\theta) = E_{regression}(S|T,\theta) + E_{regression}(T|S,\theta)$$

$$E_{regression}(S|T,\theta) = \sum_{s\in S} \frac{1}{2}\|f(v_{s\_root}) - v_{t\_root}\|^2$$

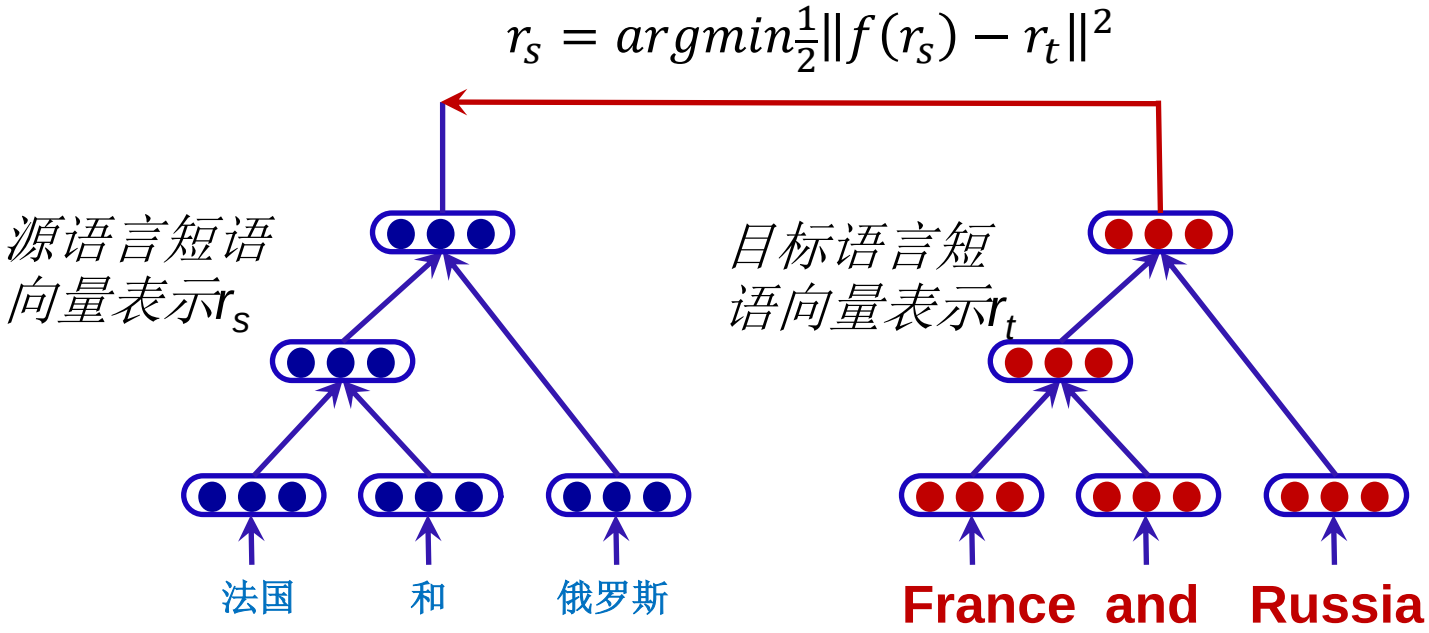
基于语义向量空间的短语表示

- 协同训练：第一步pre-training：利用无监督RAE模型分别学习源语言和目标语言短语的分布式表示



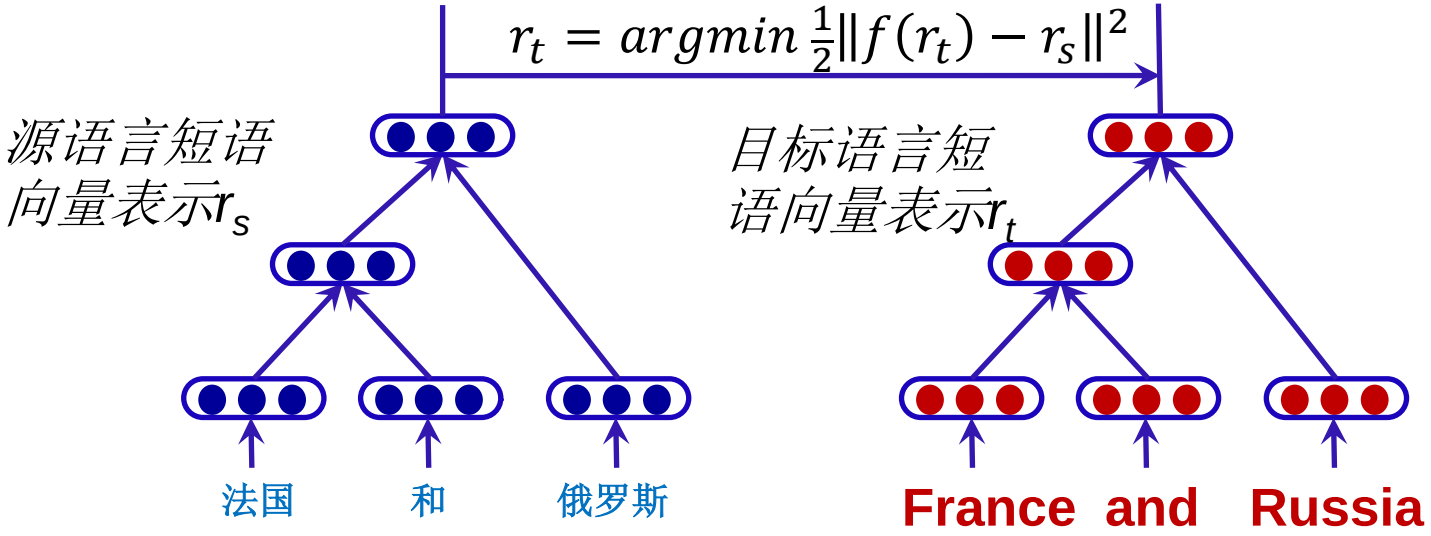
基于语义向量空间的短语表示

- 协同训练：第二步fine-tuning，1) 将目标短语表示视为源语言短语的正确语义表示，有监督地优化源语言短语的分布式表示



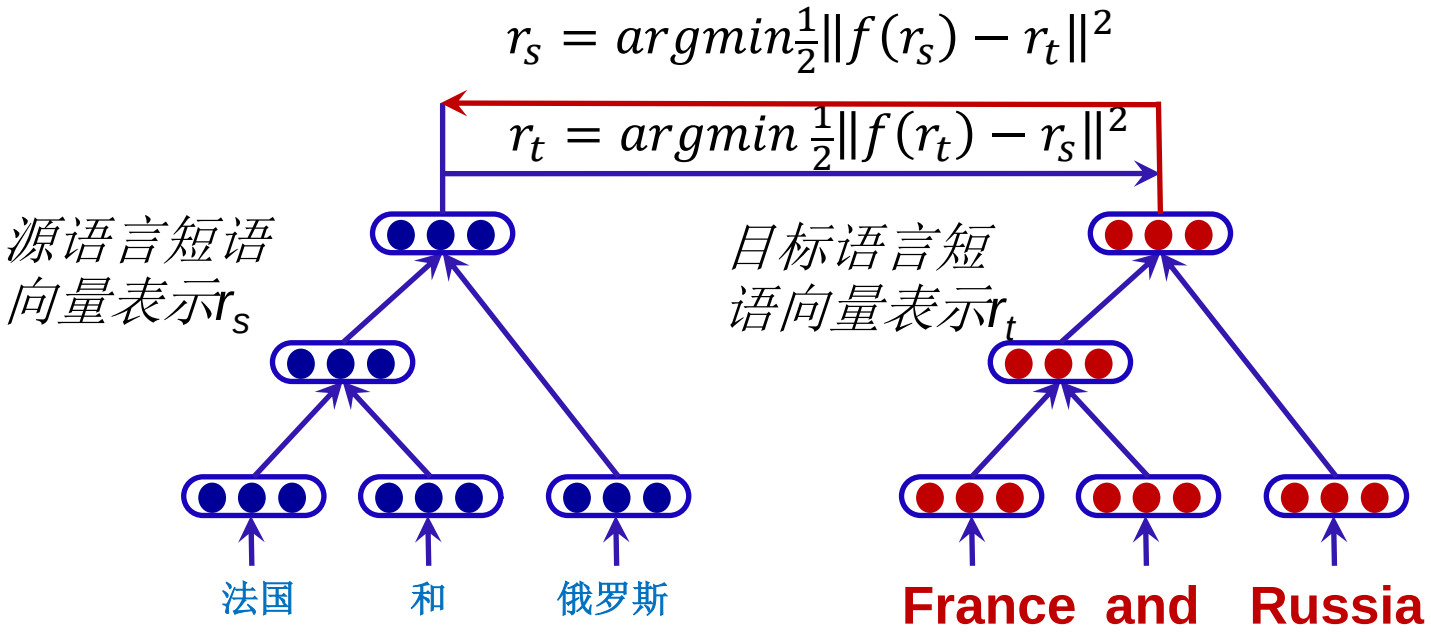
基于语义向量空间的短语表示

- 协同训练：第二步fine-tuning，2) 将源语言短语表示视为目标语言短语的正确语义表示，有监督地优化目标语言短语的分布式表示



基于语义向量空间的短语表示

- 协同训练：第三步收敛检测，若误差小于阈值或者迭代次数超过预设值，则训练结束

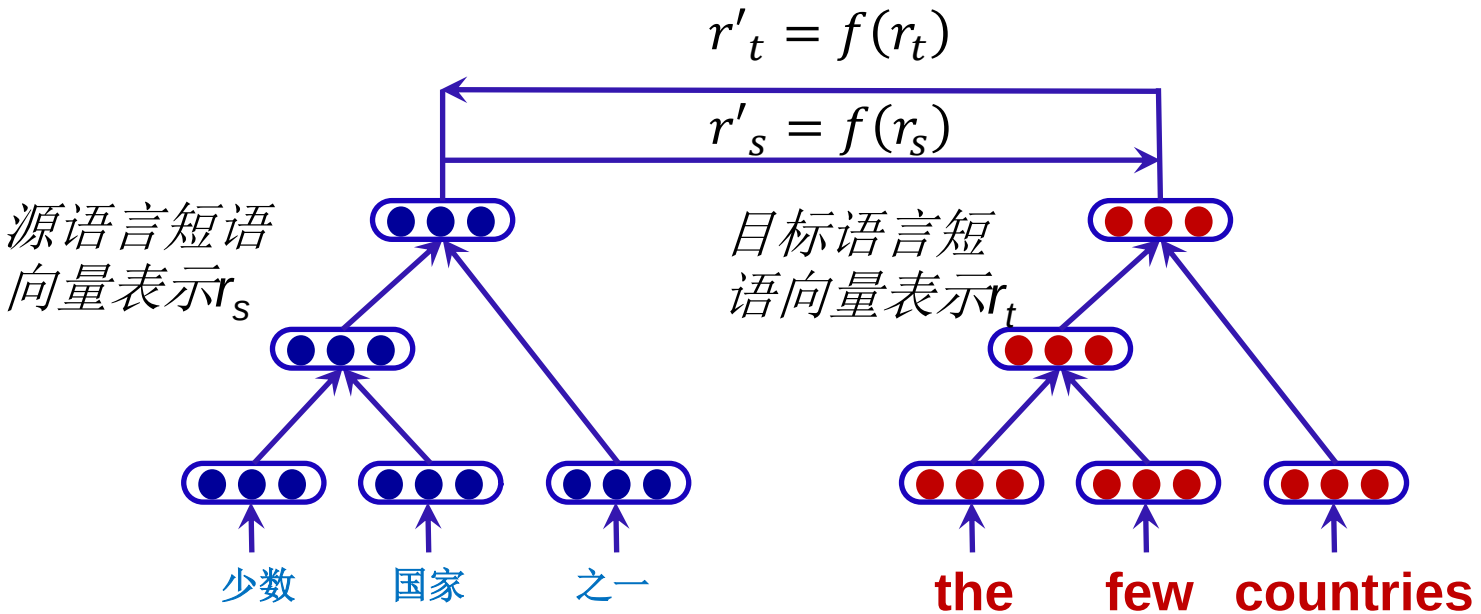


基于语义向量空间的短语表示

相似短语:	do not agree will definitely reject will never accept … …
相似短语:	abstract meaning real meaning intrinsic logic … …
相似短语:	what is your opinion what do you think about how do you view those … …

基于语义向量空间的翻译置信度估计

(少数 国家 之一, the few countries)　$\boldsymbol{p(t|s), p(s|t)}$



$$\boldsymbol{p(t|s) = sim(r_t, r'_t)}$$

$$\boldsymbol{p(t|s) = sim(r_t, r'_s)}$$

0.7 ⟵ $\begin{bmatrix}0.2\\0.2\\0.5\\0.5\\0.4\end{bmatrix}$ $\begin{bmatrix}0.5\\0.5\\0.4\\0.2\\0.5\end{bmatrix}$

基于语义向量空间的翻译置信度估计

- 汉语-英语：210万训练数据

System	NIST03	NIST04	NIST05	NIST06	NIST08	ALL
MEBTG	35.81	36.91	34.69	33.83	27.17	34.82
+2feats 50-dim	36.43 (0.62个)	37.64 (**0.73**个)	35.35 (0.66个)	35.53 (**1.70**个)	28.59 (**1.42**个)	35.84 (1.02个)
+2feats 100-dim	36.45 (**0.64**个)	37.44 (0.53个)	35.58 (0.89个)	35.42 (1.59个)	28.57 (1.40个)	36.03 (**1.21**个)
+2feats 200-dim	36.34 (0.53个)	37.35 (0.44个)	35.78 (**1.09**个)	34.87 (1.04个)	27.84 (0.67个)	35.62 (0.80个)

[Zhang et al., ACL-2014]

基于短语的统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S})$$

$$= \underset{T,S_1^K}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}, S_1^K|\boldsymbol{S})$$

$$= \underset{T,S_1^K,T_1^K,T_1^{K'}}{\operatorname{argmax}} \underline{\boldsymbol{P}(S_1^K|\boldsymbol{S})}\ \underline{\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})}\ \underline{\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)}\ \underline{\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)}$$

剩下的三个核心模型：

1. 短语翻译模型： $\boldsymbol{P}(T_1^K|S_1^K,\boldsymbol{S})$
2. 短语调序模型： $\boldsymbol{P}\left(T_1^{K'}|T_1^K,S_1^K,\boldsymbol{S}\right)$
3. 目标语言模型： $\boldsymbol{P}\left(\boldsymbol{T}|T_1^{K'},T_1^K,S_1^K,\boldsymbol{S}\right)$

最大熵短语调序模型

(与 北韩, with North Korea)　(有 邦交, have the diplomatic relations)

特征提取

f0=与, f1=北韩, f2=有, f3=邦交, f4=with, f5=korea, f6=have, f7=relations

最大熵模型

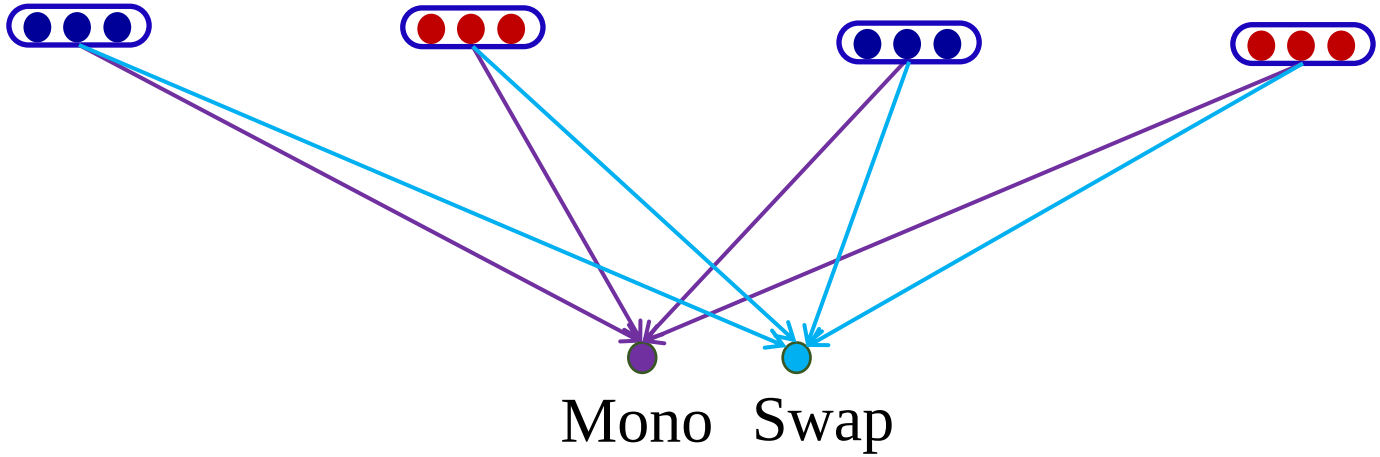
Swap

问题①：信息利用率低，仅利用短语的边界词

问题②：数据稀疏问题严重

基于语义向量空间的短语调序模型

(与 北韩, with North Korea)　(有 邦交, have the diplomatic relations)



问题①：短语表示学习　　问题②：训练目标函数

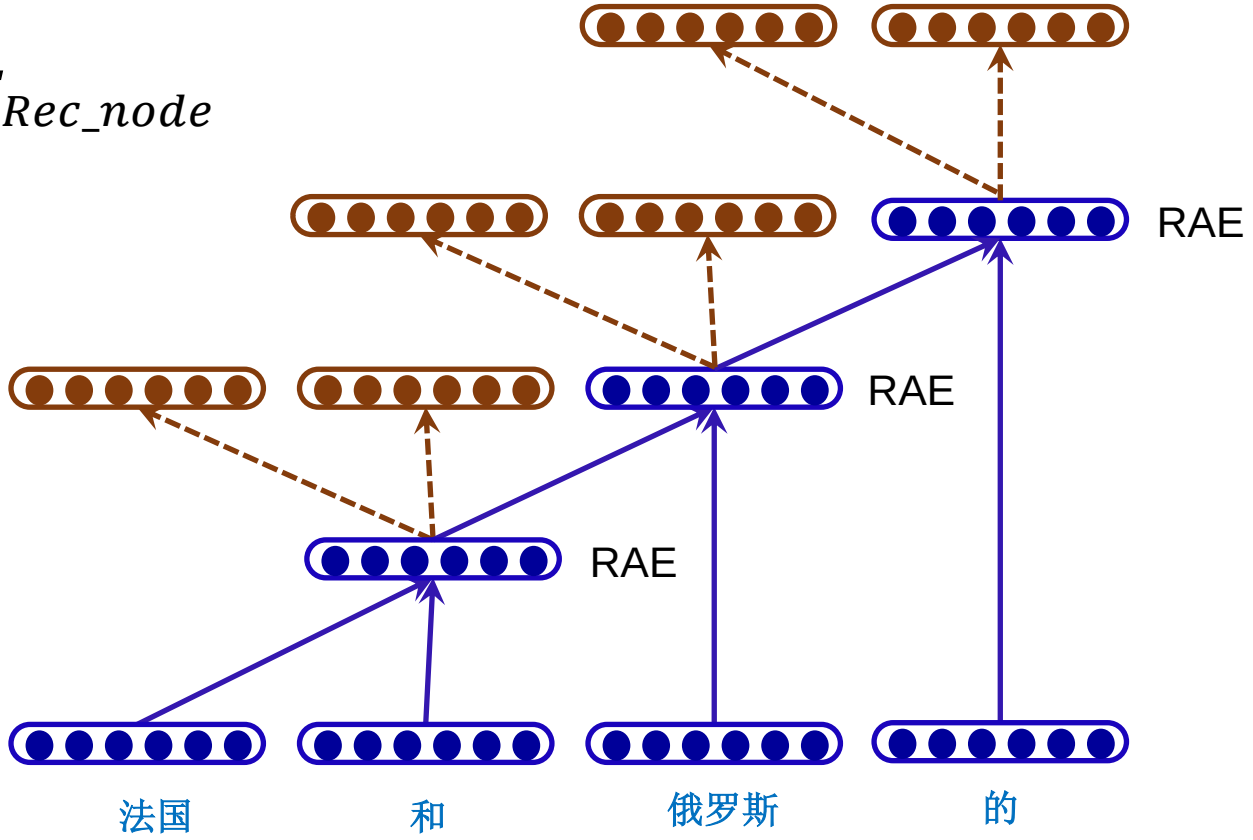
$$\boldsymbol{P}(o|X_{c1},X_{c2},X_{e1},X_{e2}) = \frac{exp\big(f(o,X_{c1},X_{c2},X_{e1},X_{e2})\big)}{\sum_{o'} exp\big(f(o',X_{c1},X_{c2},X_{e1},X_{e2})\big)}$$

$$f(o,X_{c1},X_{c2},X_{e1},X_{e2}) = f(W^o[X_{c1},X_{c2},X_{e1},X_{e2}] + b^o)$$

基于语义向量空间的短语表示

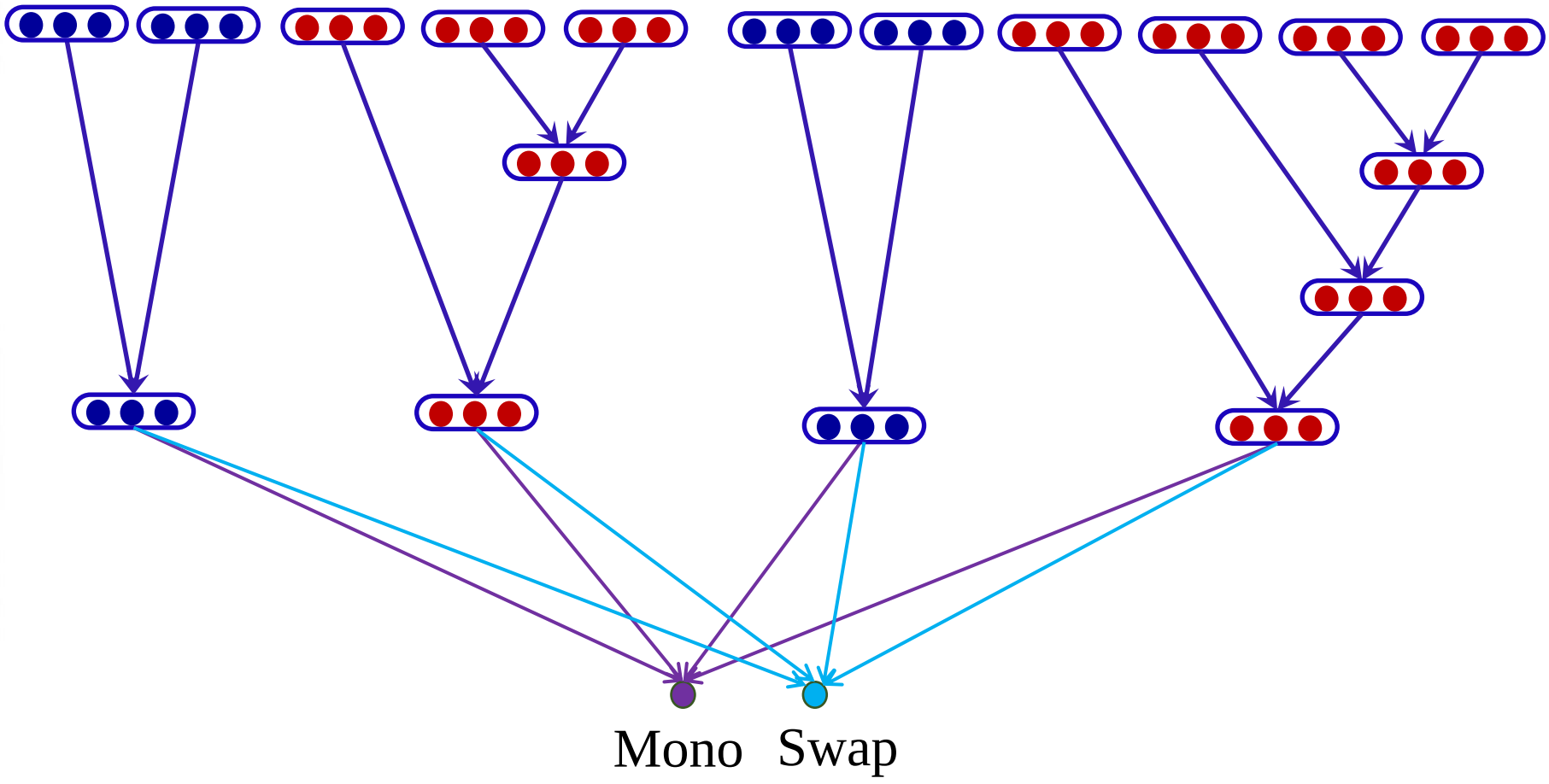
问题①：基于递归自动编码器的短语表示学习

$$E_{total} = \sum_{node} E_{Rec\_node}$$



基于语义向量空间的短语调序模型

(与 北韩, with North Korea)　(有 邦交, have the diplomatic relations)



基于语义向量空间的短语调序模型

问题②：训练目标函数 $S = \{s = (o, X_{c1}, X_{c2}, X_{e1}, X_{e2})\}$

$$J = \alpha E_{rec}(S;\theta) + (1-\alpha)E_{reo}(S;\theta) + R(\theta)$$

重构误差

调序误差

正则化项

X_{c1}

$$E_{rec} = \sum_{node \in X_{c1}} E_{rec\_node}(c_1, c_2; \theta)$$

$$E_{rec\_node}(c_1, c_2; \theta) = \frac{1}{2}\|[c_1, c_2] - [c'_1, c'_2]\|^2$$

基于语义向量空间的短语调序模型

问题②：训练目标函数 $S = \{s = (o, X_{c1}, X_{c2}, X_{e1}, X_{e2})\}$

$$J = \alpha E_{rec}(S;\theta) + (1-\alpha)E_{reo}(S;\theta) + R(\theta)$$

重构误差

调序误差

正则化项

$$E_{reo}(s;\theta) = -\sum_{o} d(o)\, log(P_o(o|X_{c1}, X_{c2}, X_{e1}, X_{e2}))$$

$$d(o) = \{0,1\}$$

基于语义向量空间的短语调序模型

cluster 1	cluster 2	cluster 3
1.18	works for	alternative duties
accessibility	verify on	one-day conference
wheelchair	tunnels from	armed groups
candies	transparency in	chinese language works
cough	opinion at	eating habits

cluster 4	cluster 5
these people who	of the three
the reasons why	on the fundamental
the story of how	over the entire
the system which	through its own
the trend towards	with the best

短语表示的效果

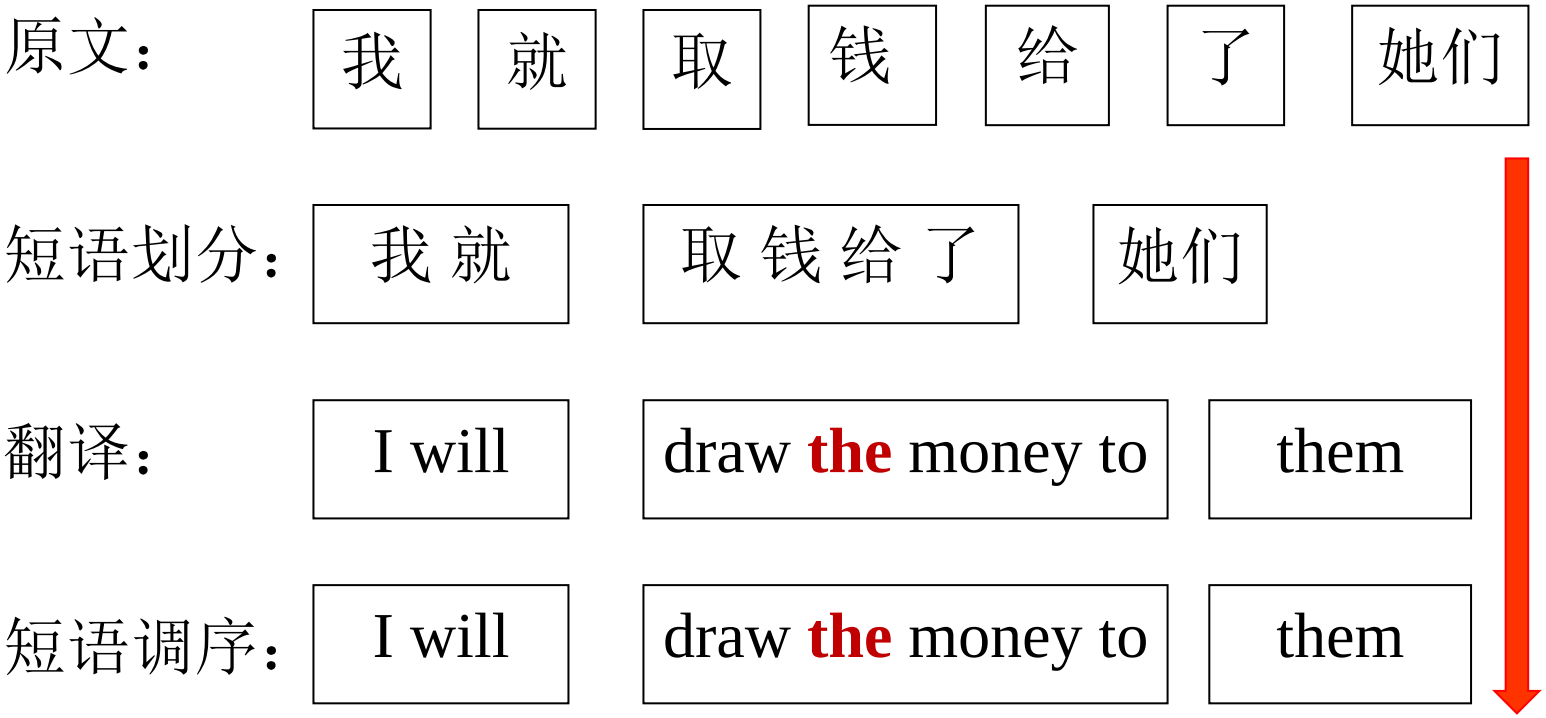
基于语义向量空间的短语调序模型

- 汉语-英语：123万双语训练数据

Setting	NIST06	NIST08
maxent	30.40	23.75
neural	**31.61**	**24.82**

[Li et al., EMNLP-2013; COLING-2014]

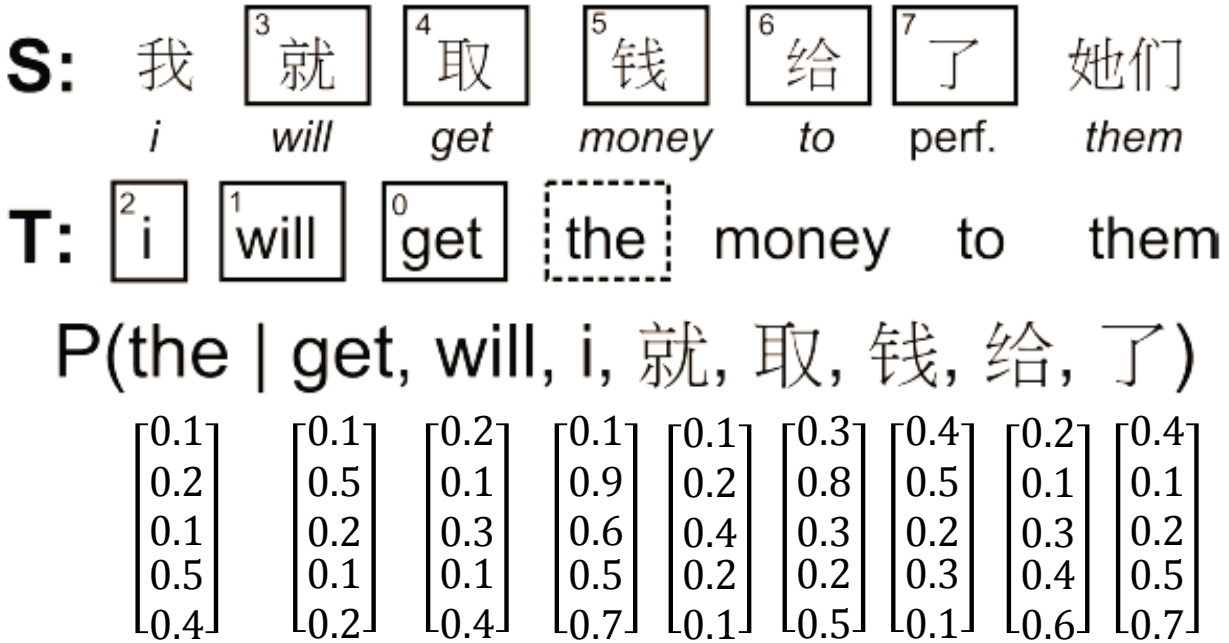
联合翻译模型与语言模型



$$P(the) \approx P(the|\text{取 钱 给 了}, draw\ the\ money\ to)$$

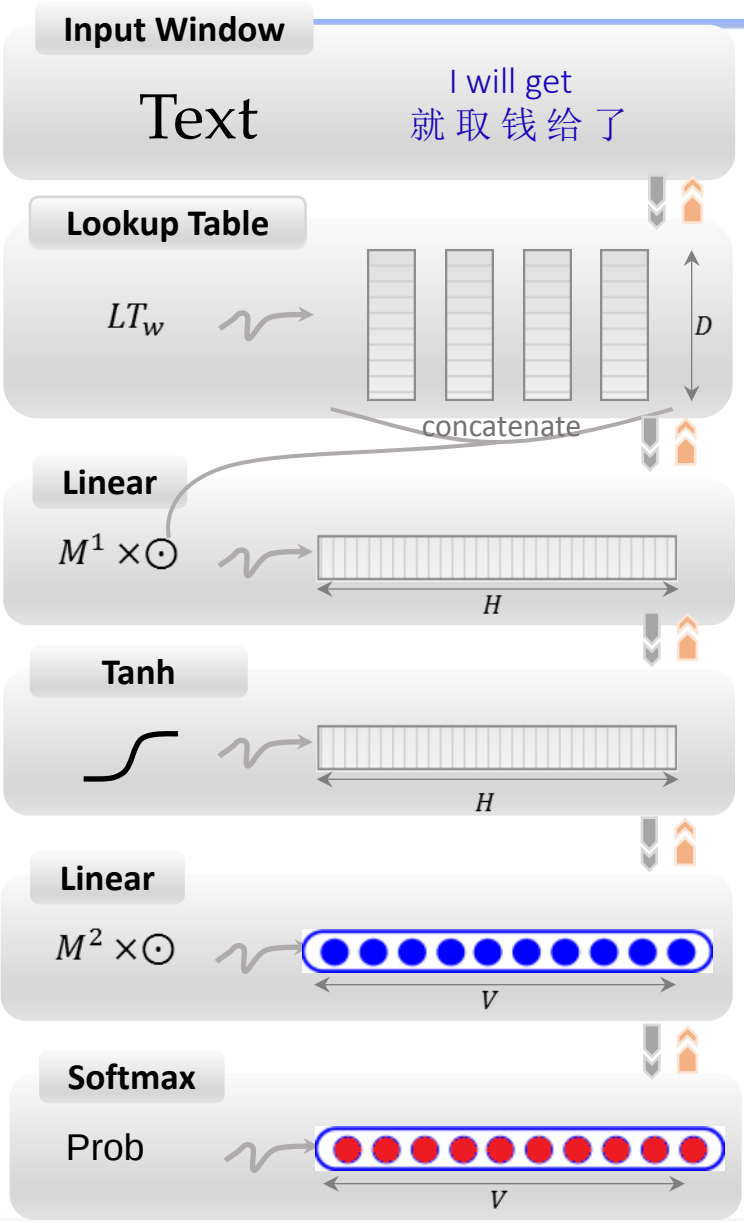
选择某个译文单词的概率仅取决于局部短语翻译规则，以及已生成的译文！

联合翻译模型与语言模型



$$P(e_i) \approx P(e_i|e_1 \cdots e_{i-1}, f)$$
$$\approx P\left(e_i|e_{i-3} \cdots e_{i-1}, f_{j-c} \cdots f_j \cdots f_{j+c}\right)$$

联合翻译模型与语言模型



- 上下文
  - 目标语言 4-gram
  - 源语言中心词左右5个词
- 词向量 (192 维)
- 两个隐藏层 (512维)
- 输出层 softmax

$$P(\boldsymbol{e_i}|e_{i-3}\cdots e_{i-1}, f_{j-c}\cdots f_j \cdots f_{j+c})$$

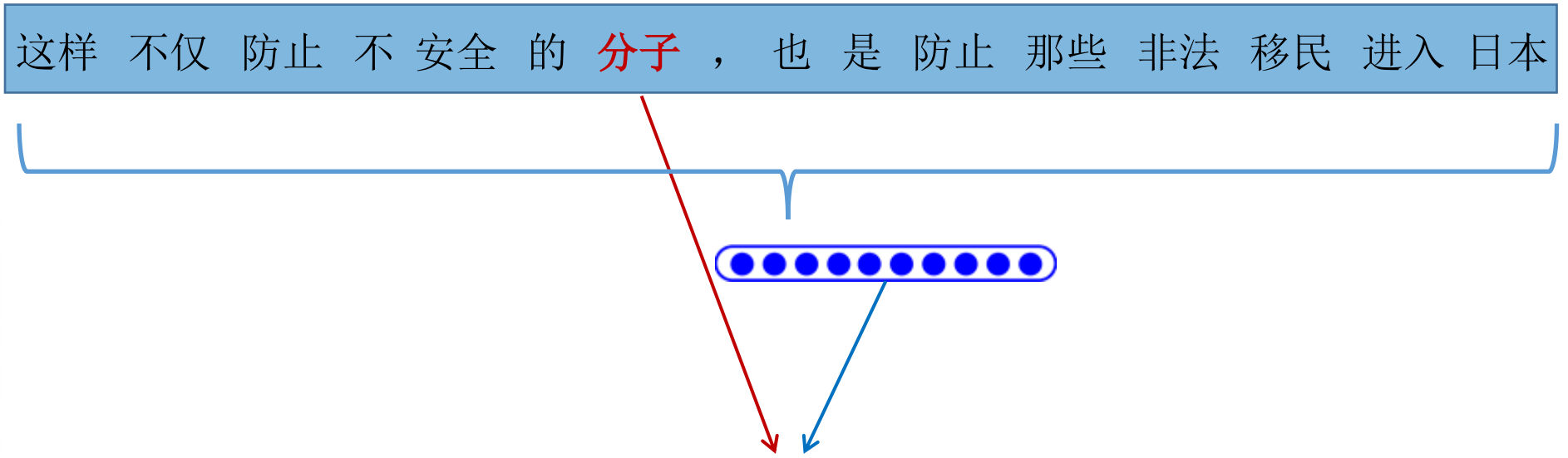
联合翻译模型与语言模型

	Ar-En	Ch-En
	BLEU	BLEU
OpenMT12 - 1st Place	49.5	32.6
OpenMT12 - 2nd Place	47.5	32.2
OpenMT12 - 3rd Place	47.4	30.8
...	...	...
OpenMT12 - 9th Place	44.0	27.0
OpenMT12 - 10th Place	41.2	25.7
Baseline (w/o RNNLM)	48.9	33.0
Baseline (w/ RNNLM)	49.8	33.4
+ S2T/L2R NNJM (Dec)	51.2	34.2
+ S2T NNLTM (Dec)	52.0	34.2
+ T2S NNLTM (Resc)	51.9	34.2
+ S2T/R2L NNJM (Resc)	52.2	34.3
+ T2S/L2R NNJM (Resc)	52.3	34.5
+ T2S/R2L NNJM (Resc)	52.8	34.7
"Simple Hier." Baseline	43.4	30.1
+ S2T/L2R NNJM (Dec)	47.2	31.5
+ S2T NNLTM (Dec)	48.5	31.8
+ Other NNJMs (Resc)	49.7	32.2

- 汉语-英语：
  NIST-2012 受限评测

[Devlin et al., ACL-2014]

句子表示学习在统计机器翻译中的应用



this not only prevents dangerous **people** … …

三个问题

Q1: 我们为什么要学习整个句子的语义表示?

这样 不仅 防止 不 安全 的 分子 ， 也 是 防止 那些 非法 移民 进入 日本

Q2: 如何学习句子的语义表示?

this not only prevents dangerous **people** … …

Q3: 如何将句子的语义表示融入统计翻译模型?

三个问题

Q1: 我们为什么要学习整个句子的语义表示?

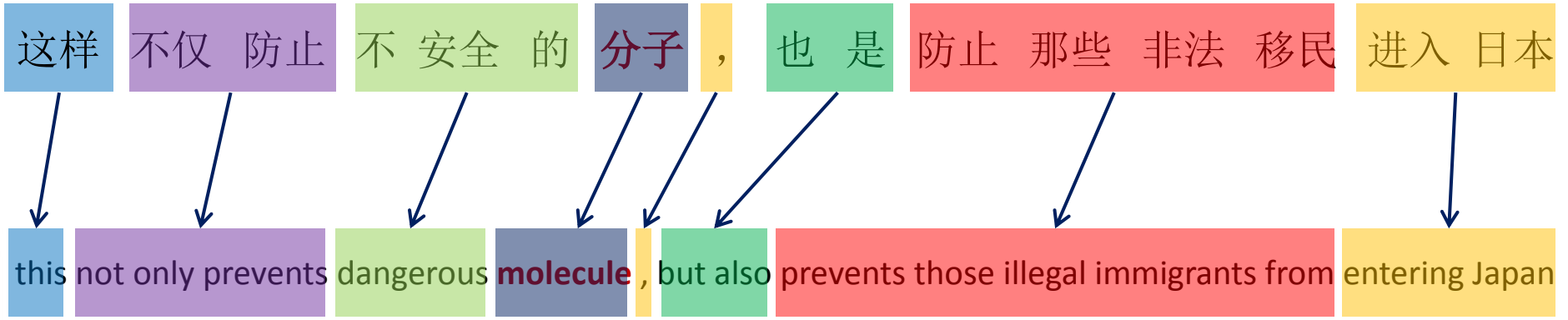
这样 不仅 防止 不 安全 的 分子 ， 也 是 防止 那些 非法 移民 进入 日本

Q2: 如何学习句子的语义表示?

this not only prevents dangerous **people** … …

Q3: 如何将句子的语义表示融入统计翻译模型?

句子表示的必要性


这样 不仅 防止 不 安全 的 分子 ， 也 是 防止 那些 非法 移民 进入 日本
this not only prevents dangerous molecule , but also prevents those illegal immigrants from entering Japan

句子表示的必要性

这样 不仅 防止 不 安全 的 分子 ， 也 是 防止 那些 非法 移民 进入 日本

this not only prevents dangerous **molecule** , but also prevents those illegal immigrants from entering Japan

三个问题

Q1: 我们为什么要学习整个句子的语义表示?

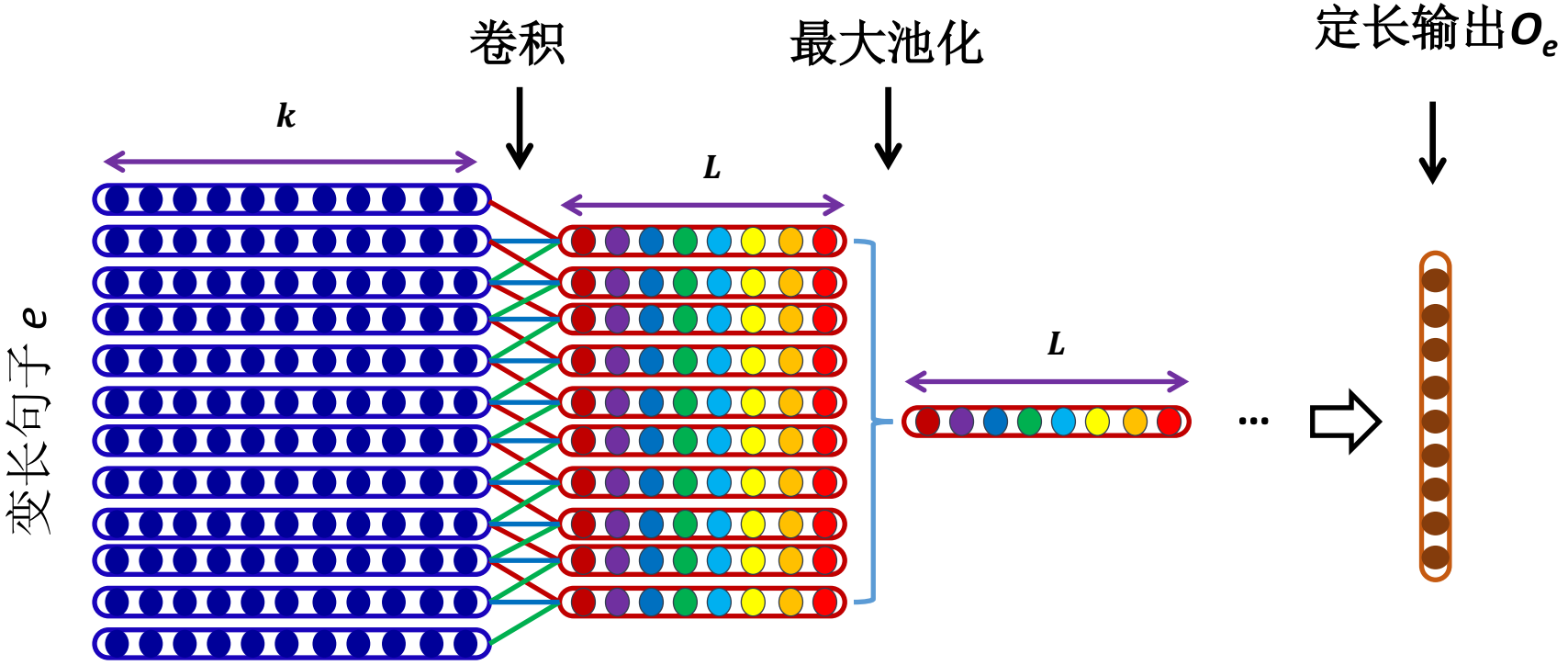
这样 不仅 防止 不 安全 的 分子 ， 也 是 防止 那些 非法 移民 进入 日本

Q2: 如何学习句子的语义表示?

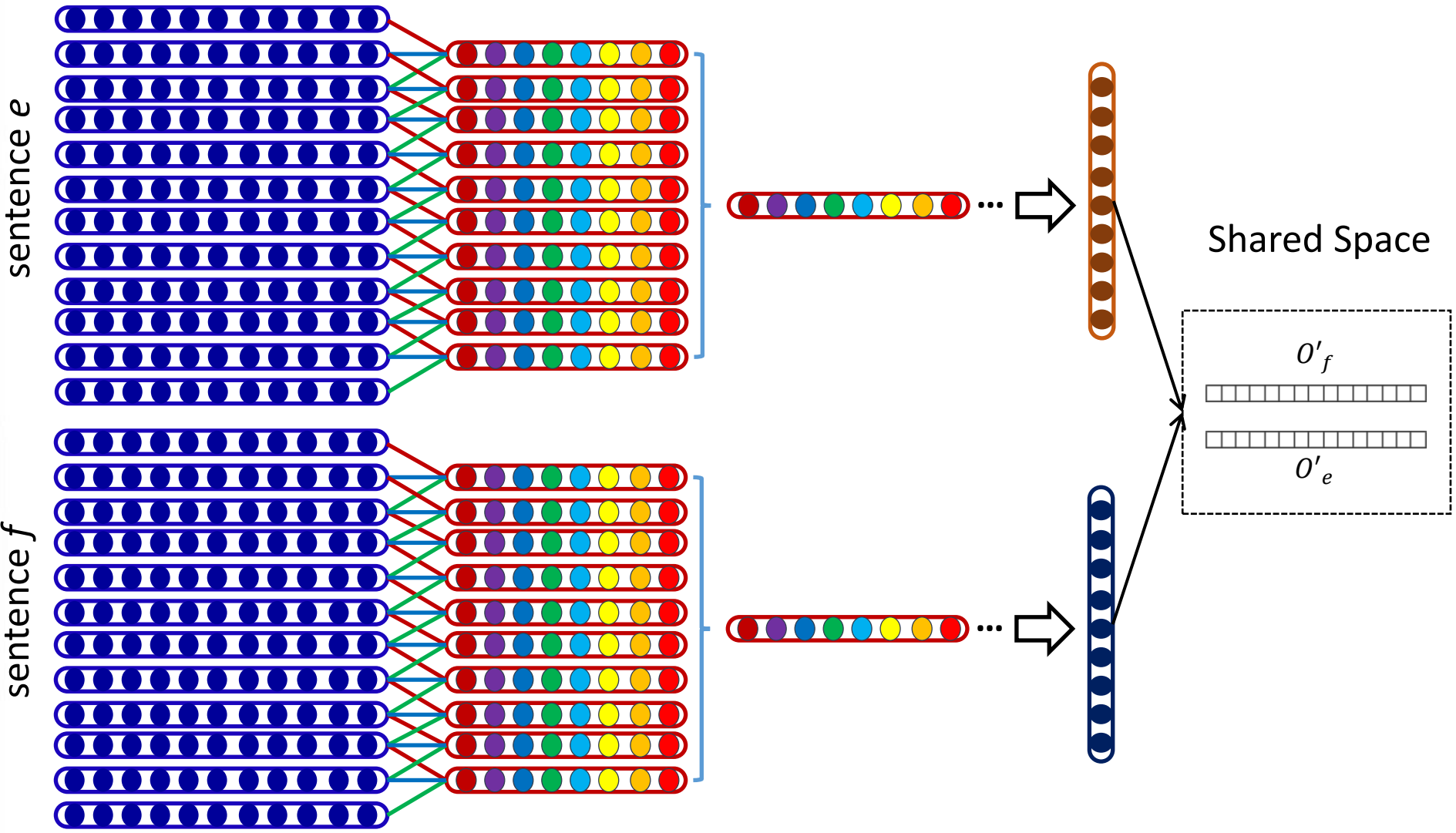
this not only prevents dangerous **people** … …

Q3: 如何将句子的语义表示融入统计翻译模型?

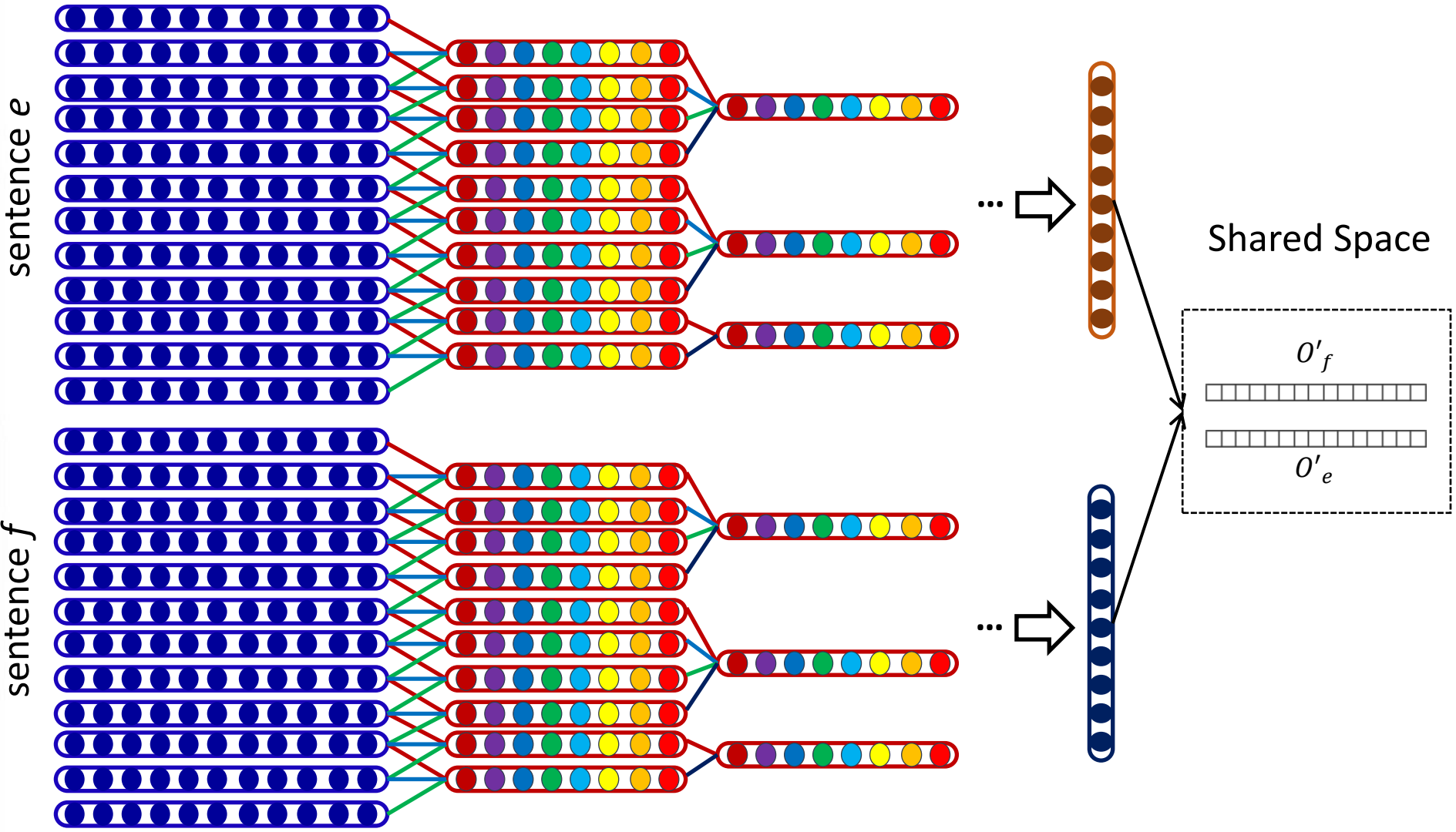
句子表示学习–卷积神经网络



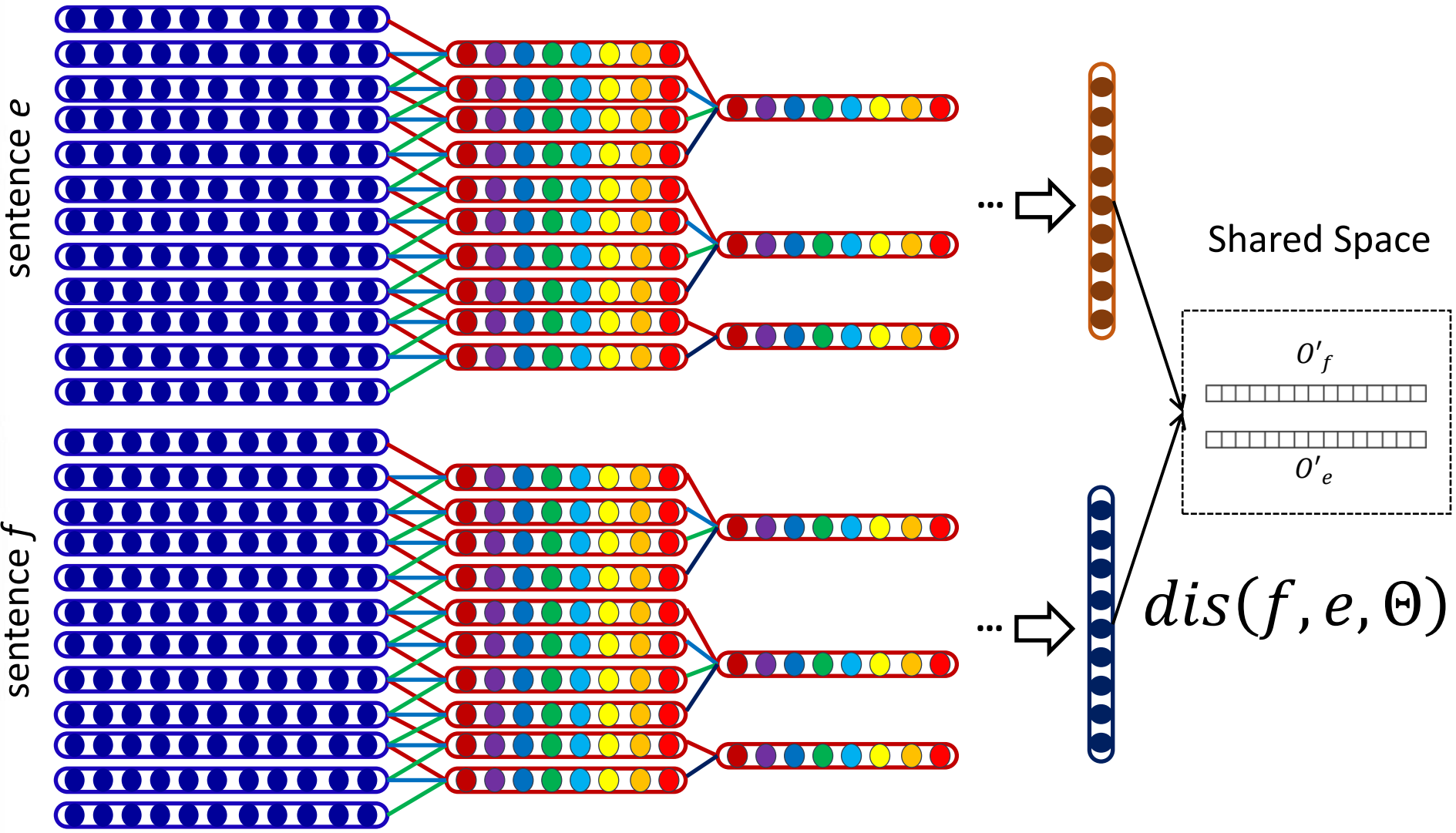
双语约束的卷积神经网络



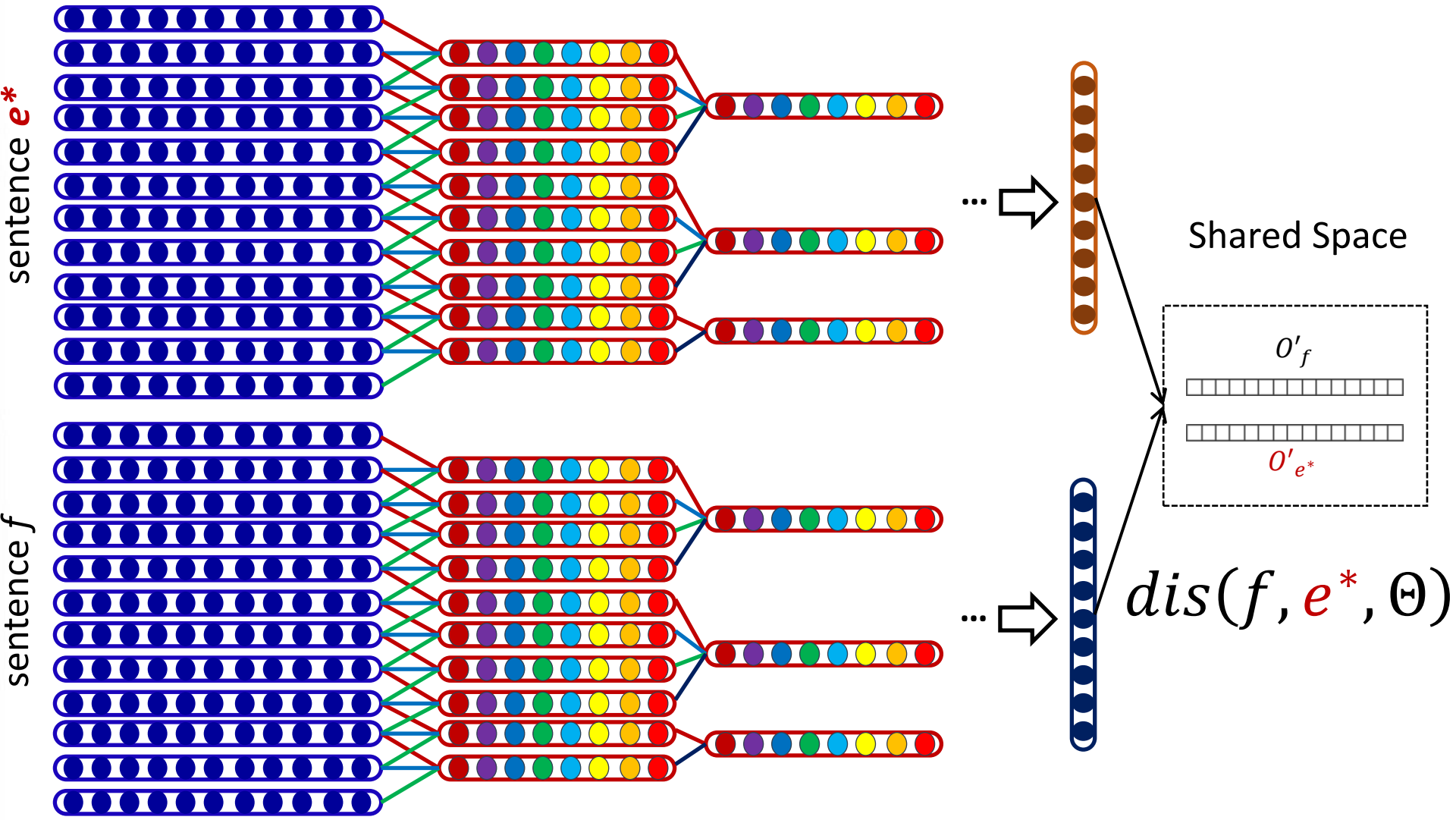
基于语块的卷积神经网络



最大间隔训练



最大间隔训练

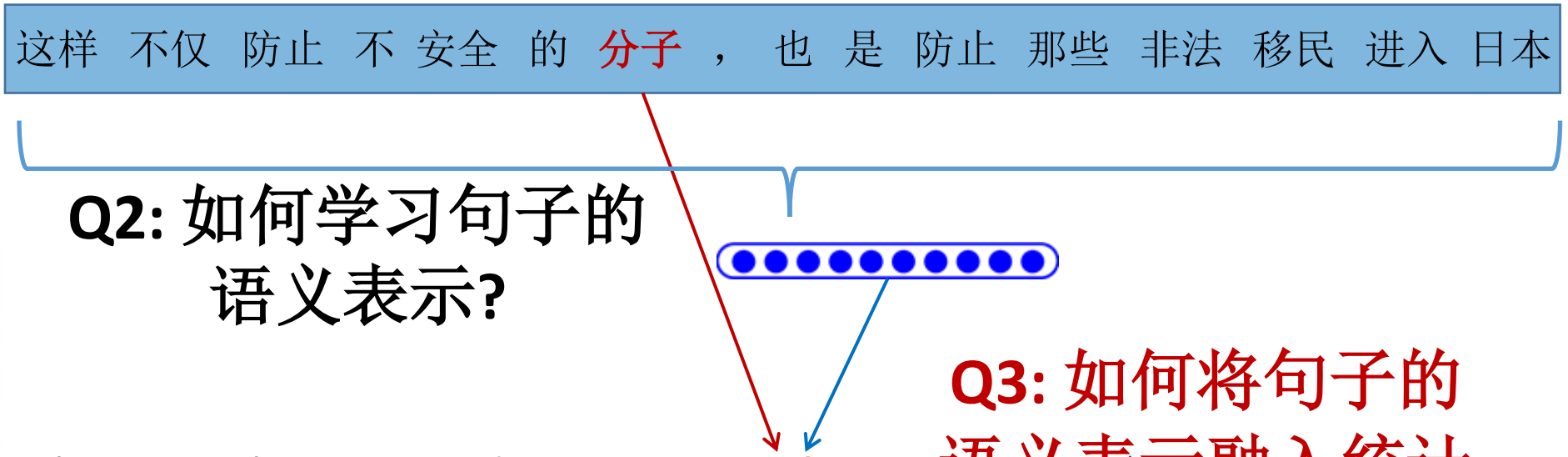


最大间隔训练

$$j(f, e, e^*, \Theta)$$
$$= max\left(0, 1 + dis(f, e, \Theta) - dis(f, e^*, \Theta)\right)$$

三个问题

Q1: 我们为什么要学习整个句子的语义表示?



Q2: 如何学习句子的语义表示?

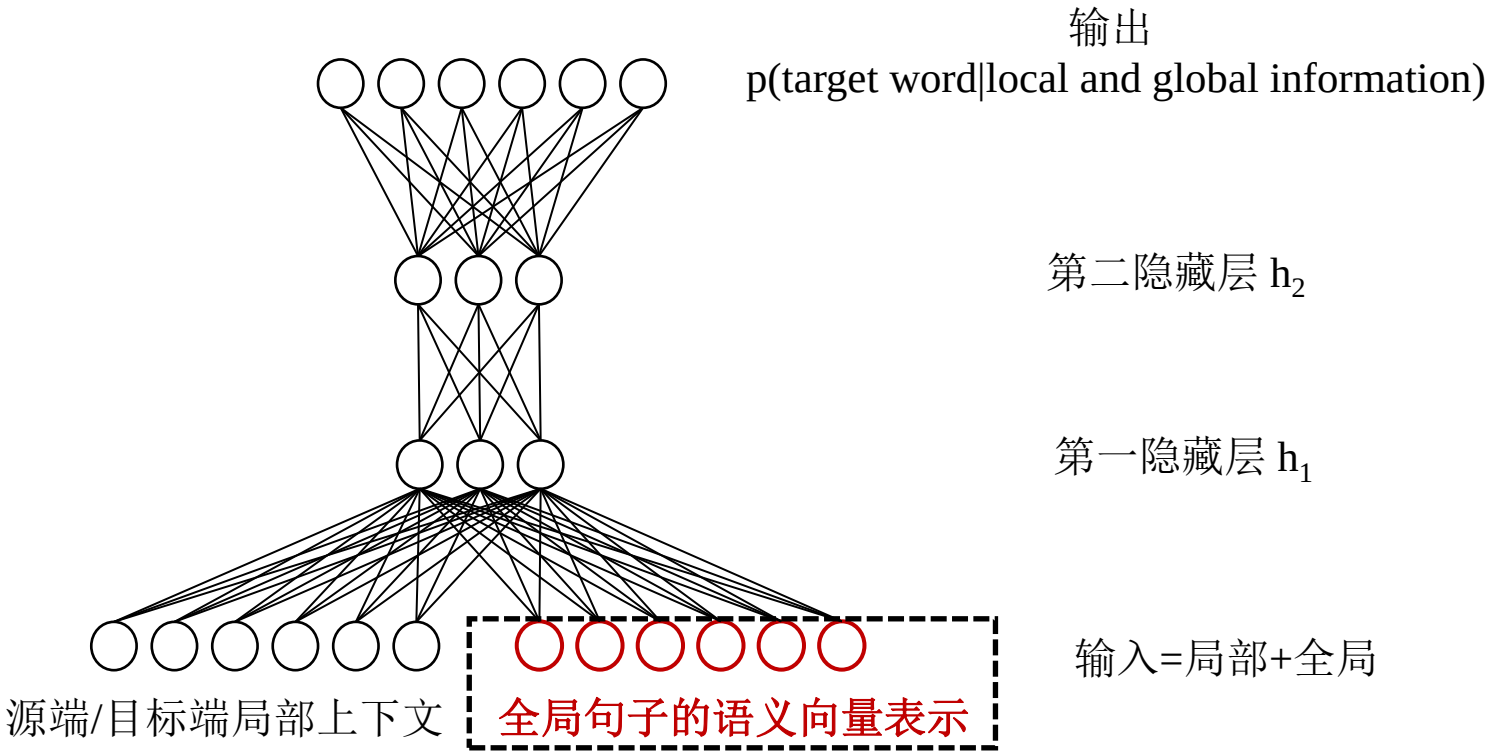
Q3: 如何将句子的语义表示融入统计翻译模型?

基线系统

$$P(e_i) \approx P(e_i|e_1 \cdots e_{i-1}, f)$$
$$\approx P\left(e_i|e_{i-3} \cdots e_{i-1}, f_{j-c} \cdots f_j \cdots f_{j+c}\right)$$

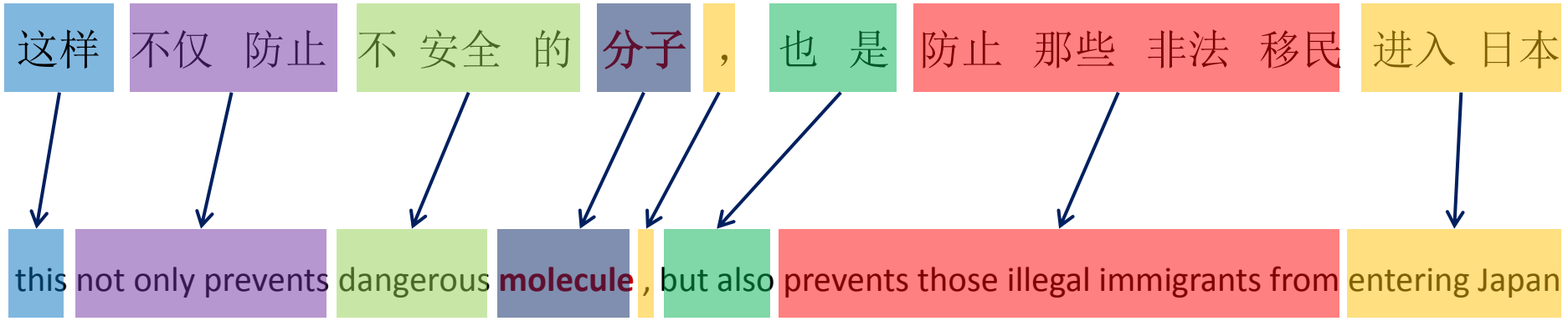
卷积神经网络–统计机器翻译

$$P(e_i) \approx P\left(e_i | e_{i-3} \cdots e_{i-1}, f_{j-c} \cdots f_j \cdots f_{j+c}\right)$$
$$\approx P\left(e_i | e_{i-3} \cdots e_{i-1}, f_{j-c} \cdots f_j \cdots f_{j+c}, \boldsymbol{f}\right)$$



融入统计机器翻译

系统: 层次短语翻译系统

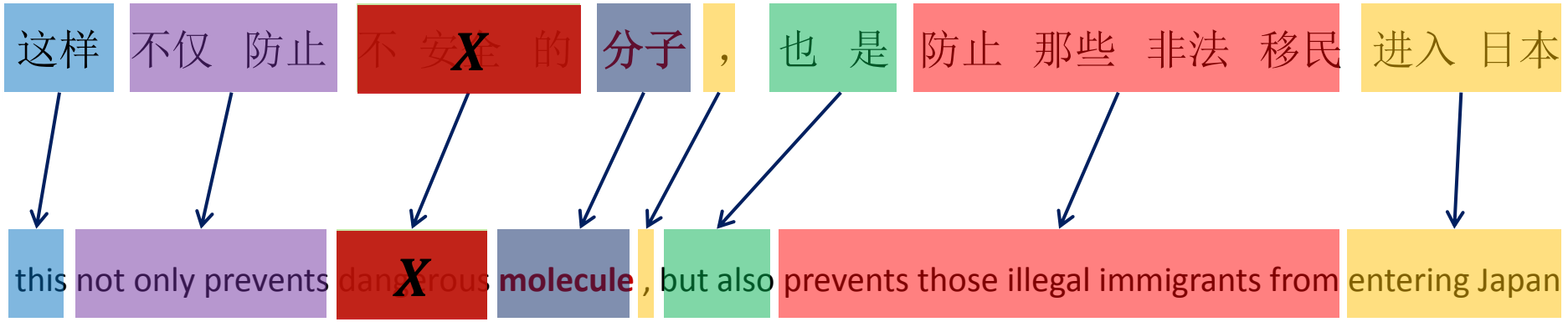


$X = \langle$不 安全 的， $\mathrm{da}ngerous\rangle$

$X = \langle X_0$ 分子，$X_0\ molecule\rangle$

融入统计机器翻译

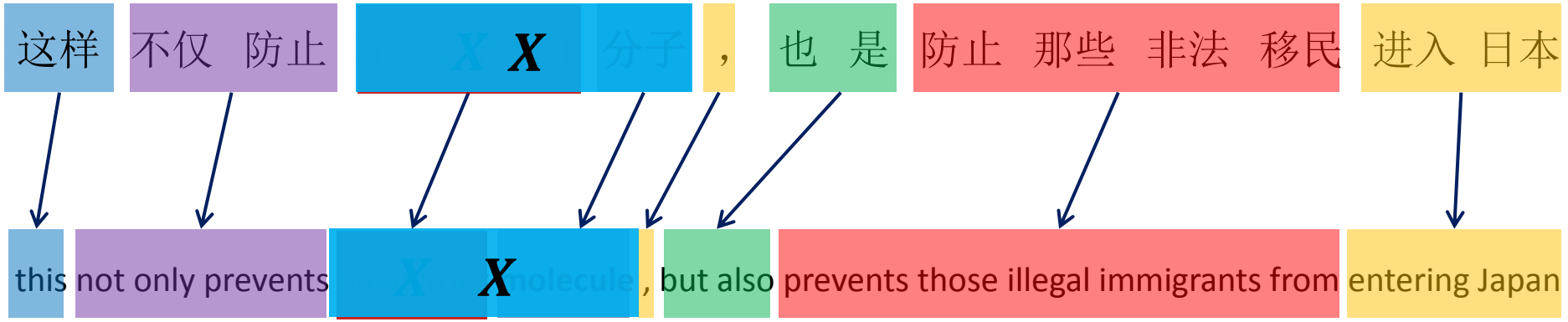
系统: 层次短语翻译系统



$X = \langle$不 安全 的， $dangerous\rangle$

融入统计机器翻译

系统: 层次短语翻译系统



$X = \langle$不 安全 的， $dangerous\rangle$

$X = \langle X_0$ 分子，$X_0\ molecule\rangle$

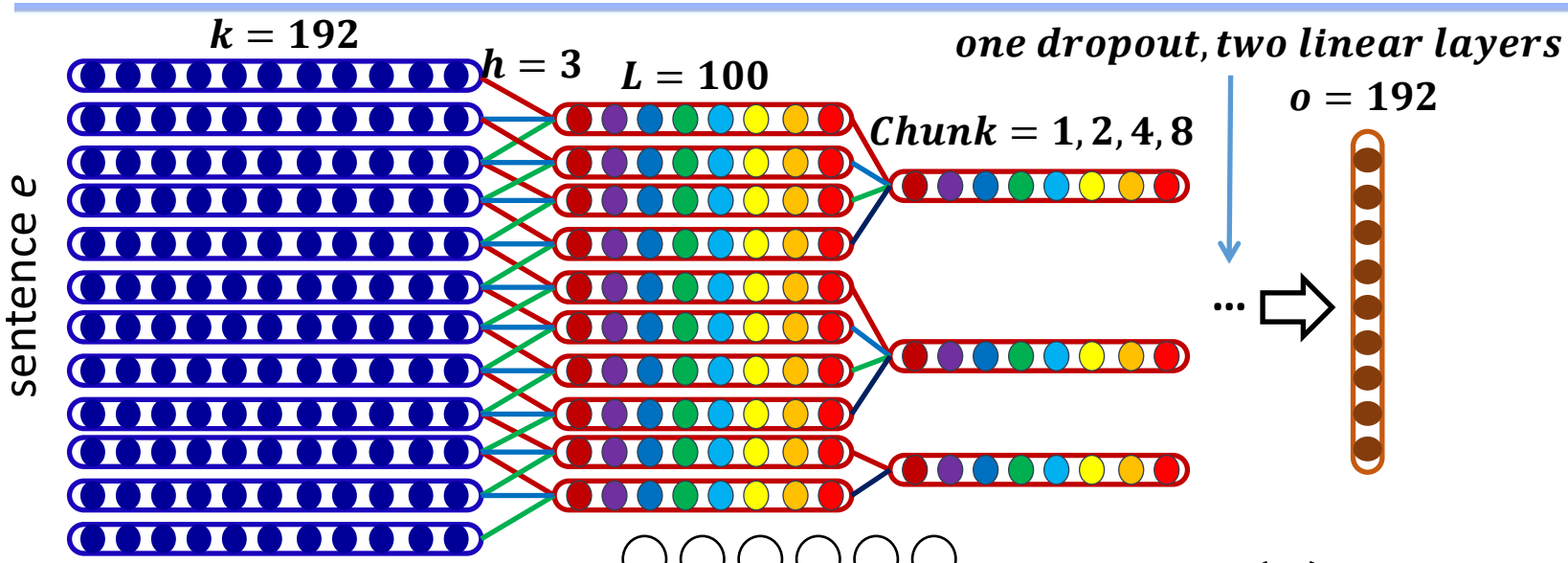
$X = \langle$不 安全 的 分子， $dangerous\ molecule\rangle$

Log-linear Model: $P(e_i|e_{i-3}\cdots e_{i-1}, f_{j-c}\cdots f_j\cdots f_{j+c})$

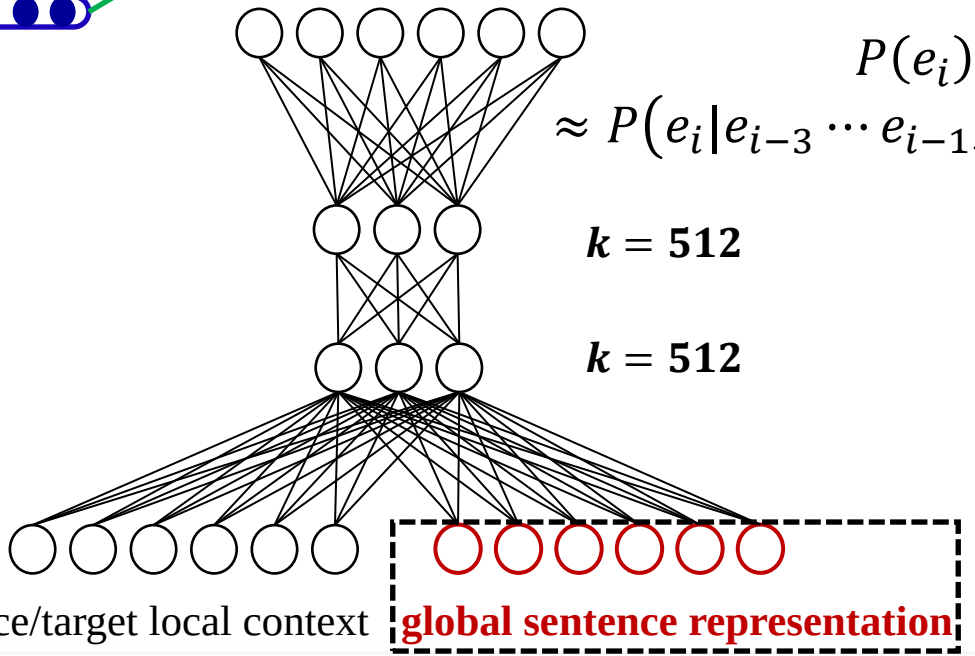
正向反向概率　　目前语言模型特征　　神经网络翻译概率

正向反向词汇化概率　　规则数目和译文长度特征

网络参数



$$P(e_i) \approx P\left(e_i | e_{i-3} \cdots e_{i-1}, f_{j-c} \cdots f_j \cdots f_{j+c}, \boldsymbol{f}\right)$$



句子表示在统计机器翻译中的效果

- 汉语-英语：210万双语训练数据

System	MT03	MT05	MT06	MT08
HPB	35.98	34.66	35.25	27.80
+NNJM	36.93	35.55^{+}	35.77	28.64^{+}
+AVE_SENT	37.16	35.88^{+}	36.07^{+}	29.19^{+}
+BCCNN-1	37.32	36.06^{+}	36.42^{+}	29.35^{+*}
+BCCNN-2	37.75	36.24^{+}	36.65^{+*}	29.97^{+*}
+BCCNN-4	**37.98**	36.22^{+}	$\mathbf{36.78}^{+*}$	$\mathbf{30.02}^{+*}$
+BCCNN-8	37.64	$\mathbf{36.29}^{+*}$	36.49^{+}	29.98^{+*}

0.95↑

1.38↑

[Zhang et al., IJCAI-2015]

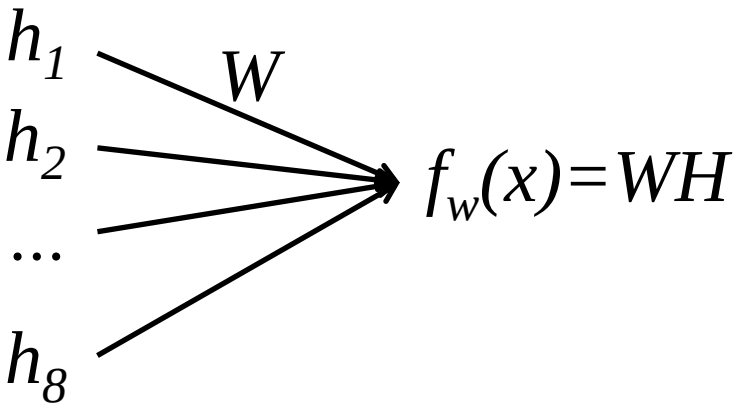
统计机器翻译

$$T' = \underset{T}{\operatorname{argmax}} P(T|S) = \underset{T}{\operatorname{argmax}} \left\{ \sum_{1}^{M} \lambda_m h_m(T,S) \right\}$$

$$h_1(T,S) = \log p(t|s)$$
$$h_2(T,S) = \log p(s|t)$$
$$h_3(T,S) = \log p_{lex}(t|s)$$
$$h_4(T,S) = \log p_{lex}(s|t)$$
$$h_5(T,S) = \log P\left(T_1^{K'}|T_1^K, S_1^K, S\right)$$
$$h_6(T,S) = \log P\left(T|T_1^{K'}, T_1^K, S_1^K, S\right)$$
$$h_7(T,S) = \log len(T)$$
$$h_8(T,S) = \log count(phrases) = \log K$$

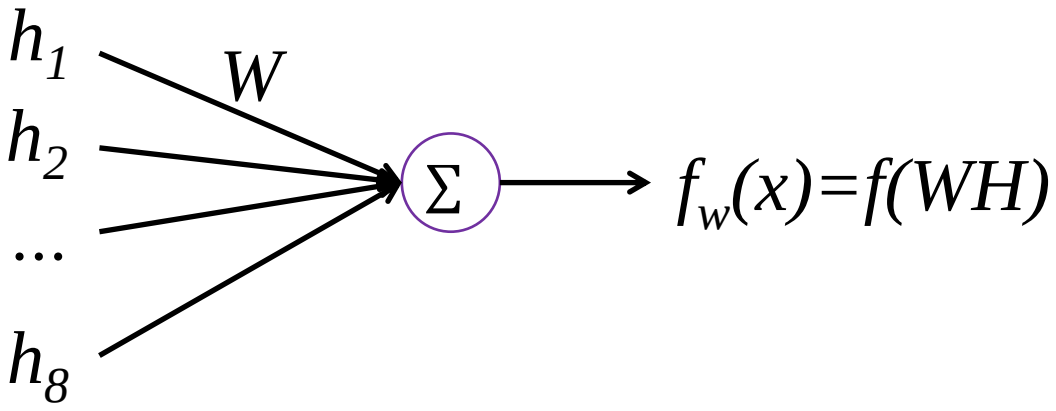
统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S}) = \underset{T}{\operatorname{argmax}} \left\{ \sum_{\mathbf{1}}^{\boldsymbol{M}} \boldsymbol{\lambda_m h_m}(\boldsymbol{T}, \boldsymbol{S}) \right\}$$



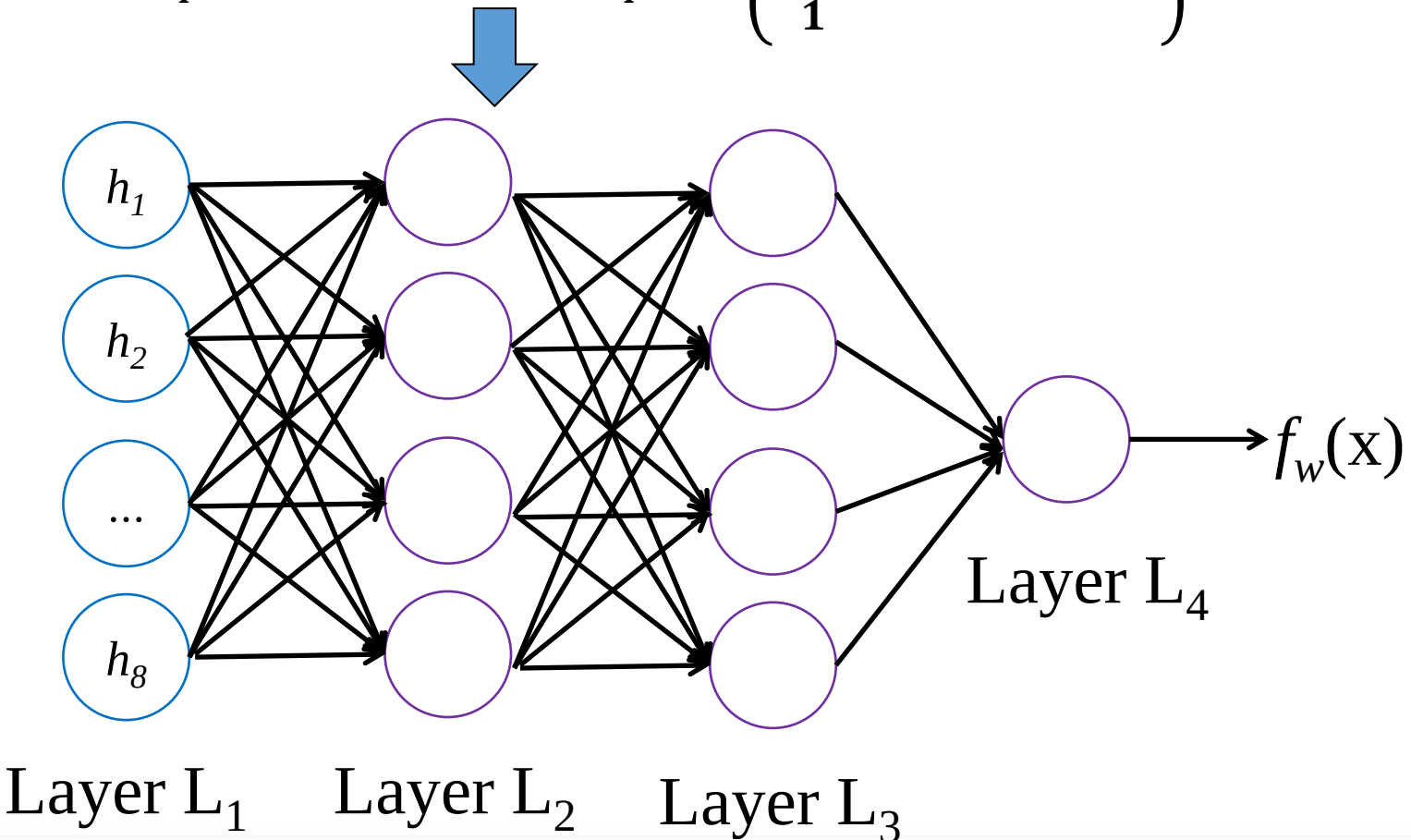
非线性统计机器翻译

$$\boldsymbol{T}' = \underset{T}{\operatorname{argmax}} \boldsymbol{P}(\boldsymbol{T}|\boldsymbol{S}) = \underset{T}{\operatorname{argmax}} \left\{ \sum_{\mathbf{1}}^{\boldsymbol{M}} \boldsymbol{\lambda_m h_m}(\boldsymbol{T}, \boldsymbol{S}) \right\}$$



非线性统计机器翻译

$$T' = \underset{T}{\operatorname{argmax}} P(T|S) = \underset{T}{\operatorname{argmax}} \left\{ \sum_{1}^{M} \lambda_m h_m(T,S) \right\}$$



非线性统计机器翻译

- 汉语-英语：820万双语训练数据

Criteria	MT03(train)	MT02(dev)	MT04	MT05
BR_c	35.02	36.63	34.96	34.15
BR	38.66	40.04	38.73	37.50
BW	39.55	39.36	38.72	37.81
PW	38.61	38.85	38.73	37.98

[Huang et al., ACL-2015]

参考文献

1. Yoshua Bengio, Rejean Ducharme, Pascal Vincent and Christian Jauvin. A neural probabilistic language model. *Journal of Machine Learning Research*, 3:1137–1155..
2. David Chiang. Hierarchical phrase-based translation. *Computational Linguistics*, 2007.
3. Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. Fast and robust neural network joint models for statistical machine translation. *In Proc. of ACL* 2014.
4. Jianfeng Gao, Xiaodong He, Yih Wen-tao and Li Deng. Learning continuous phrase representations for translation modeling. *In Proc. of ACL 2014.*
5. Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. *In Proc. of ACL* 2007.
6. Peng Li, Yang Liu, and Maosong Sun. Recursive autoencoders for itg-based translation. *In Proc. of EMNLP* 2013.
7. Peng Li, Yang Liu, Maosong Sun, Tatsuya Izuha and Dakun Zhang. A neural reordering model for phrase-based translation. *In Proc. of COLING* 2014.
8. Fandong Meng, Zhengdong Lu, Mingxuan Wang, Hang Li, Wenbiing Jiang and Qun Liu. Encoding source language with convolutional neural network for machine translation. *In Proc. of ACL* 2015.
9. Franz Josef Och and Hermann Ney. Discriminative training and maximum entropy models for statistical machine translation. *In Proc. of ACL* 2002.
10. Shujian Huang, Huadong Chen, Xinyu Dai and Jiajun Chen. Non-linear Learning for Statistical Machine Translation. *In Proc. of ACL 2015.*

参考文献

11. Jinsong Su, Deyi Xiong, Biao Zhang, Yang Liu, Junfeng Yao and Min Zhang. Bilingual Correspondence Recursive Autoencoder for Statistical Machine Translation. *In Proc. of EMNLP 2015*.

12. Akihiro Tamura, Taro Watanabe and Eiichiro Sumita. Recurrent neural networks for word alignment model. *In Proc. of ACL* 2014.

13. Ashish Vaswani, Yinggong Zhao, Victoria Fossum and David Chiang. Decoding with large-scale neural language models improves translation. *In Proc. of EMNLP* 2013.

14. Deyi Xiong, Qun Liu and Shouxun Lin. Maximum entropy based phrase reordering model for statistical machine translation. *In Proc. of ACL* 2006.

15. Nan Yang, Shujie Liu, Mu Li, Ming Zhou and Nenghai Yu. Word alignment modeling with context dependent deep neural network. *In Proc. of ACL* 2013.

16. Jiajun Zhang, Shujie Liu, Mu Li, Ming Zhou and Chengqing Zong. Bilingually-constrained phrase embeddings for machine translation. *In Proc. of ACL* 2014.

17. Jiajun Zhang, Dakun Zhang and Jie Hao. Local Translation Prediction with Global Sentence Representation. *In Proc. of IJCAI* 2015.

18. Jiajun Zhang and Chengqing Zong. Deep Neural Networks in Machine Translation: an Overview. *IEEE Intelligent Systems* 2015.

19. Will Y Zou, Richard Socher, Daniel Cer and Christopher D Manning. Bilingual word embeddings for phrase-based machine translation. *In Proc. of EMNLP* 2013.

开源工具

1, Google Word2Vec, http://code.google.com/p/word2vec/

2，NPLM，前馈神经网络，便于在统计机器翻译中应用前馈神经网络语言模型，http://nlg.isi.edu/software/nplm/

3，RWTHLM，前馈和循环神经网络，便于在统计机器翻译中应用循环神经网络语言模型，http://www-i6.informatik.rwth-aachen.de/web/Software/index.html

……

谢谢!

Q&A