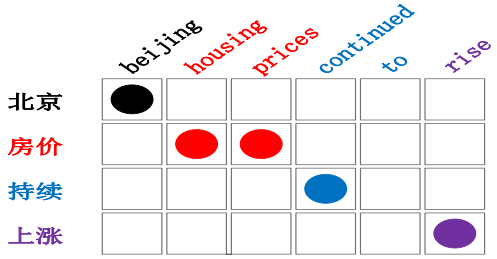
第四讲 短语翻译表构造 --短语抽取

学习目标: 深刻理解基于短语统计机器翻译系统中短语[1]抽取的基本算法，可自行开发短语抽取模块。

短语抽取是统计机器翻译系统构建的第三步，经过第二步词对齐后，获得双语平行句对间的词对齐信息。短语抽取是短语翻译表构造的第一步[2]，短语抽取的任务是从含有词对齐信息的双语平行句对中学习解码器在翻译过程中使用的翻译短语。

本讲学习内容:

- **示例短语**



示例短语	
北京	beijing
房价	housing prices
持续	continued to
上涨	rise
北京 房价	beijing housing prices
持续 上涨	continued to rise

图 1 双语数据与词对齐信息中抽取示例短语对

短语抽取的本质是从含有词对齐信息的平行句对中抽取基于短语的统计机器翻译系统中使用的翻译短语。图 1 中展示了从含有词对齐信息的双语平行句对(图 1 中上方图所示)中抽取的短语对(图 1 中下方的“示例短语”所示)。从图 1 中可以看出，在给定词对齐信息的双语平行句对中，理想情况下是可以抽取与词对齐保持一致的短语对，如“示例短语”中所示的短语对。

- **一致性短语**

使用含有词对齐信息的双语平行句对进行短语抽取时，抽取出的短语对需要与词对齐保持一致。下面给出一致性的定义。

---

[1] 此处**短语**指代**短语对**

[2] 短语翻译表构造的第二步为**短语翻译表概率估计**

一致性定义：给定句对$(\mathbf{f}_1^J, \mathbf{e}_1^I)$与词对齐 $\mathcal{A}$，短语对$(\overline{\mathbf{f}_{j_1}^{j_2}}, \overline{\mathbf{e}_{i_1}^{i_2}})$与 $\mathcal{A}$保持一致，当且仅当$\overline{\mathbf{f}_{j_1}^{j_2}}$中的所有词汇在 $\mathcal{A}$ 中所对应的词汇在$\overline{\mathbf{e}_{i_1}^{i_2}}$范围之内，$\overline{\mathbf{e}_{i_1}^{i_2}}$中的所有词汇在 $\mathcal{A}$ 中所对应的词汇在$\overline{\mathbf{f}_{j_1}^{j_2}}$范围之内；与此同时，在$\overline{\mathbf{f}_{j_1}^{j_2}}$与$\overline{\mathbf{e}_{i_1}^{i_2}}$中，至少有一个词汇在 $\mathcal{A}$ 中。

$\left(\overline{f_{j_1}^{j_2}}, \overline{e_{i_1}^{i_2}}\right)$与 $\mathcal{A}$ 保持一致 $\Leftrightarrow$

$$\forall e_i \in \overline{e_{i_1}^{i_2}} : (f_j, e_i) \in A \Rightarrow f_j \in \overline{f_{j_1}^{j_2}}$$

$$\text{AND} \quad \forall f_j \in \overline{f_{j_1}^{j_2}} : (f_j, e_i) \in A \Rightarrow e_i \in \overline{e_{i_1}^{i_2}}$$

$$\text{AND} \quad \exists e_i \in \overline{e_{i_1}^{i_2}}, f_j \in \overline{f_{j_1}^{j_2}} : (f_j, e_i) \in A$$

从一致性的定义可以看出，图 1 中所示的 6 条示例短语对均与词对齐保持一致。

- **短语抽取算法**

在定义“一致性短语”后，本节给出在含有词对齐信息的双语平行数据中抽取所有满足“一致性”定义短语对的算法。

➢ 算法 1 的核心思想：长度从 1 到 I 遍历目标语端词串并且在源语端找到与之匹配的最小词串。如果目标语端词串中所有词汇在词对齐中对应的项都在与之匹配的源语词串范围内，并且源语端词串中所有词汇在词对齐中所对应的项都在目标语词串范围内时，同时源语、目标语词串不能只包含对空词汇，此时找到的源语端、目标语端词串便与词对齐保持一致，称双语端词串对为短语对。

算法 1 详细解释如下：

1. 算法第 1 行与第 2 行在目标语端进行二重循环，目的是遍历目标语端所有可能出现的短语；算法第 2 行设置当前目标语端词串的起始位置。
2. 算法第 3 行设置当前目标语端词串的结束位置。
3. 算法第 4 行设置当前源语端词串的起始位置

算法 1: 短语抽取算法

输入: 含有词对齐信息 $\mathcal{A}$ 的句对(f_1^J, e_1^I)
输出: 抽取的短语集合，这里以 $\mathcal{P}$ 指代

```
1:  for len = 1 to I do
2:      for i1 = 1 to I do
3:          i2 = i1 + len
4:          (j1, j2) = (J, 0)
5:          for all (i, j) ∈ A do
6:              if (i1 ≤ i ≤ i2) then
7:                  j1 = min(j, j1)
8:                  j2 = max(j, j2)
9:              end if
10:         end for
11:         add extract(j1, j2, i1, i2) to set P
12:     end for
13: end for
```

与结束位置。起始位置设置为源语句子长度的最大值，结束位置设置为 0。该设置可快速判断是否可找到与词对齐保持一致的短语。

4. 算法第 5-10 行确保目标语端词串$\overline{e_{i_1}^{i_2}}$中的所有词汇在词对齐中对应的词汇在源语端词串$\overline{f_{j_1}^{j_2}}$范围内。
5. 算法第 11 行使用 extract(j_1, j_2, i_1, i_2)函数对找到的可能短语对进行验证和扩展，确保找到短语对与词对齐保持一致。

➢ 算法 2 中 extract 函数是算法 1 中第 11 行使用的对找到的短语进行验证和扩展的函数。在与词对齐保持一致的短语对的扩展过程中，主要是短语对中源语端与目标语端边缘对空词汇的扩展。根据一致性的定义，边缘对空词汇不会影响短语一致性的性质，同时，抽取更多边缘对空扩展对空短语可获得更多上下文信息、可适当缓解词对齐不精确带来的问题。

算法 2 详细解释如下:

1. 算法第 1-3 行保证找到的源语端词串中至少有一个词汇在词对齐中对应的项在目标语词串$\overline{e_{i_1}^{i_2}}$内。
2. 算法第 4-8 行确保源语端词串$\overline{f_{j_1}^{j_2}}$中的所有词汇在词对齐中对应的词汇在目标语端词串$\overline{e_{i_1}^{i_2}}$范围内，即找到与词对齐一致的短语对。
3. 算法第 9 行初始化短语对集合为空。
4. 算法第 10-18 行扩展与词对齐保持一致的短语对，如果找到的短语对的源语端或目标语端的边界词汇对空，则扩展该短语对，将新短语对加入到短语对集合 $\mathbb{E}$ 中。
5. 算法第 19 行返回短语对集合 $\mathbb{E}$。

算法 2： extract(j_1, j_2, i_1, i_2)

输入：j_1, j_2, i_1, i_2
输出：抽取的短语集合，这里以 $\mathbb{E}$ 指代
1: if $(j_2 = 0)$ then
2: return $\emptyset$
3: end if
4: for all $(i, j) \in A$ do
5: if $(i < i_1 \ or \ i > i_2)$ then
6: return $\emptyset$
7: end if
8: end for
9: $\mathbb{E} = \emptyset$
10: $j_{start} = j_1$
11: repeat
12: $j_{end} = j_2$
13: repeat
14: add $\left(\overline{f_{j_{start}}^{j_{end}}}, \overline{e_{i_1}^{i_2}}\right)$ to set $\mathbb{E}$
15: $j_{end} ++$
16: until j_{end} aligned
17: $j_{start} --$
18: until j_{start} aligned
19: return $\mathbb{E}$

- 短语抽取流程

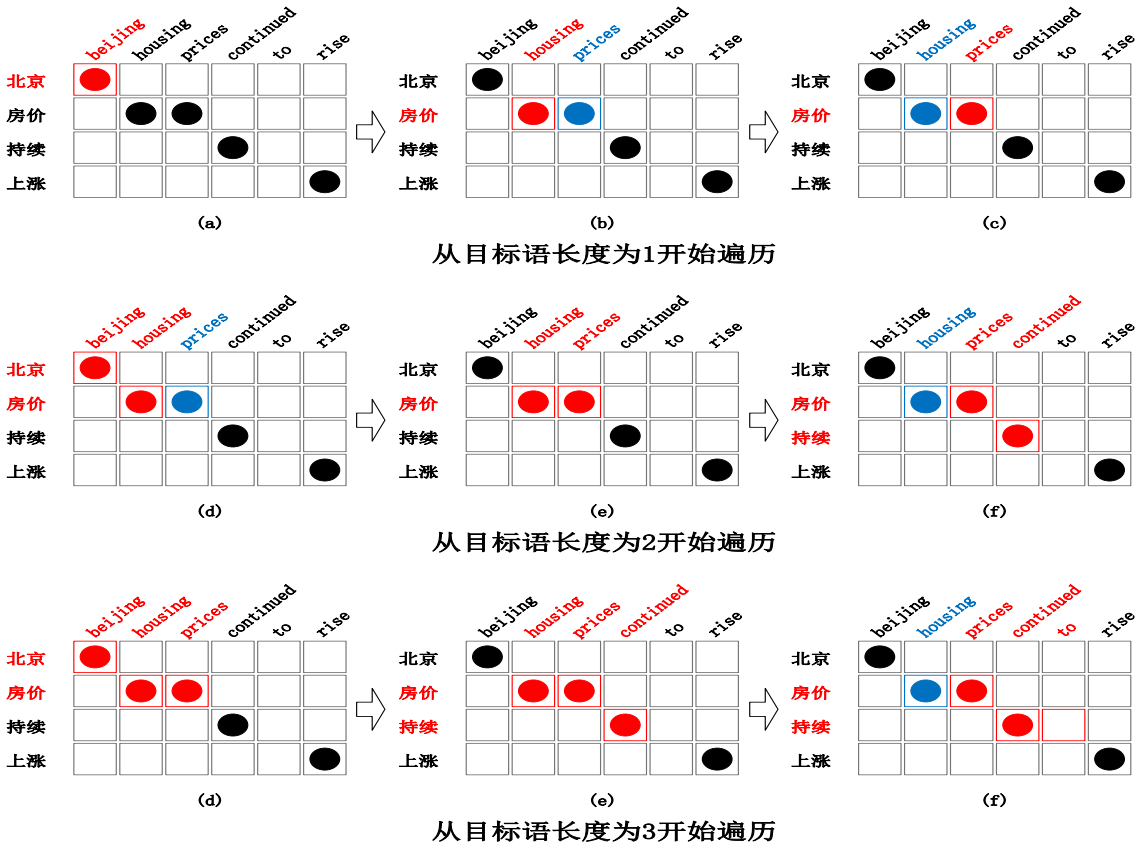
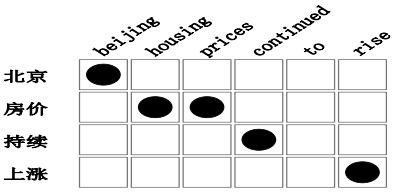


图 2 短语抽取流程，其中蓝色点为异常点

图 2 展示应用短语抽取算法 1、2 抽取短语对的基本流程。其中“从目标语长度 n 开始循环”为算法 1 的第 1 行；每组三个图如“(a),(b)和(c)”为算法 1 中第 2 行至第 12 行。图（2）中（a）图展示抽取的一个与词对齐保持一致的短语对“北京,beijing”;(b)图展示一个非法的短语对“房价,housing”，其中目标语端词汇“prices”为异常点。

- 满足一致性的短语



一致性短语

北京	beijing
持续	continued
上涨	rise
房价	housing prices
持续	continued to
上涨	to rise
北京 房价	beijing housing prices
房价 持续	housing prices continued
持续 上涨	continued to rise
北京 房价 持续	beijing housing prices continued
房价 持续	housing prices continued to
北京 房价 持续	beijing housing prices continued to
房价 持续 上涨	housing prices continued to rise
北京 房价 持续 上涨	beijing housing prices continued to rise

图 3 双语数据与词对齐信息中抽取满足一致性定义的所有短语对，以算法 1 实际短语抽取顺序排序

图 3 中所示“一致性短语”为根据上文“一致性”定义及“算法 1、2”从示例含有词对齐信息的双语平行句对中抽取的所有与词对齐保持一致的短语对。

- **参考资源**

1. NiuTrans 源码：http://www.nlplab.com/NiuPlan/NiuTrans.ch.html 下载 NiuTrans 源码包， NiuTrans 源码包目录 /NiuTrans/src/NiuTrans.PhraseExtractor/ 下， phrasetable_extractor.cpp 与 phrasetable_extractor.h 两文件实现短语抽取功能。

2. Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge, UK: Cambridge University Press.

3. Tong Xiao, Jingbo Zhu, Hao Zhang and Qiang Li. 2012. *NiuTrans: An Open Source Toolkit for Phrase-based and Syntax-based Machine Translation*. In *Proc. of ACL 2012*, page 19-24.