2020-2021学年秋季学期

自然语言处理

Natural Language Processing

授课教师： 胡玥

助　教： 于静

自然语言处理

Natural Language Processing

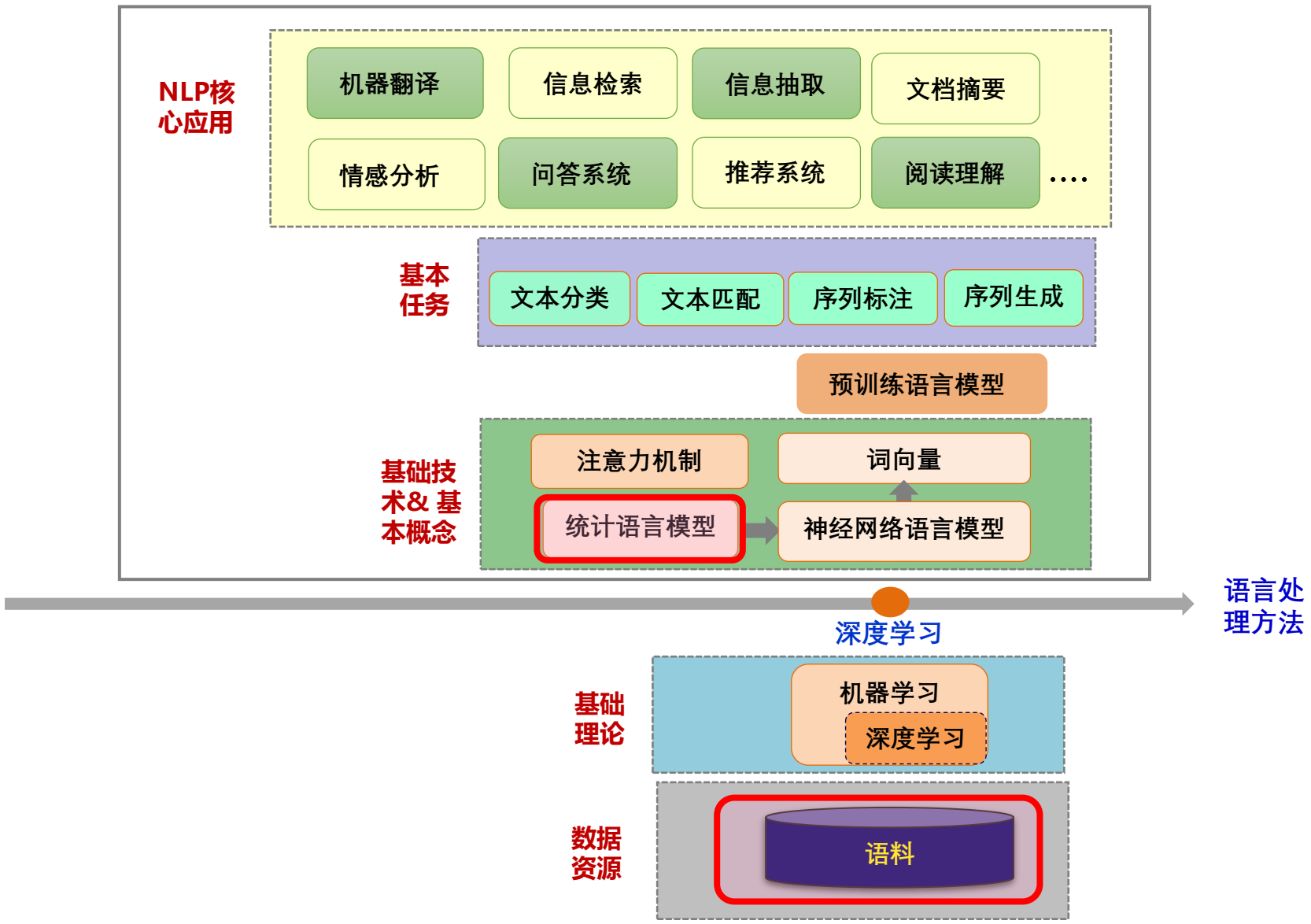
第 3 章　统计语言模型

授课教师：　胡玥

授课时间：2020.9

课程编码 201M4005H　课程名称 自然语言处理　授课团队名单 胡玥 、于静

基于深度学习的自然语言处理课程内容



第 3 章　统计语言模型

概 要

本章主要内容：介绍统计语言模型的基本概念，模型参数的学习方法，模型评价指标 和 语言模型的用途，并简要介绍几种语言模型的变形形式

本章教学目的：掌握语言模型的基本概念，理解其在语言处理中的作用，了解语言模型的变形形式：前向-后向语言模型，Skipping 语言模型

内 容 提 要

3.1　语言模型基本概念

3.2　语言模型参数估计

3.3　语言模型性能评价

3.4　语言模型应用

3.1 语言模型基本概念

问题引入： （语音识别问题）

下表中，给定拼音串，如何确定对应的文字?

拼音串（无声调）	ni xian zai zai gan shen mo
候选字串	你 线 在 再 干 什 么
	你 现 在 在 干 什 么
	尼 先 在 在 感 什 么
	……
候选词串	你 现在 在 感什么
	你 现在 在 干什么
	你 先在 再 干什么
	……
正确文字串	你现在在干什么

如何判断语句合理性?

传统规则法：

句子是否合乎语法、语义（语言学分析）

问题：

判断过程复杂

其他方式?

3.1 语言模型基本概念

弗莱德里克·贾里尼克

语言模型提出

弗莱德里克·贾里尼克（美国工程院院士）

提出了用数学的方法描述语言规律（语言模型）

语言模型思想

用句子$S=w_1,w_2,\ldots,w_n$ 的概率 p(S) 刻画句子的合理性。（统计自然语言处理的基础模型）

对语句合理性判断：

规则法：判断是否合乎语法、语义（语言学定性分析）

统计法： 通过可能性（概率）的大小来判断（数学定量计算）

3.1 语言模型基本概念

句子概率 p(S)定义：

自然语言为上下文相关的信息传递方式

语句 s = w_1 w_2 … w_n 的 概率p(S)定义为：

$$p(S)=p(w_1)p(w_2|w_1)...p(w_n|w_1,...,w_{n-1}) = \prod_{i=1}^{n} p(w_i \mid w_1 \cdots\cdots w_{i-1})$$

语言模型

其中：当 i=1 时，$p(w_1|w_0) = p(w_1)$

语言模型

$$p(S) = \prod_{i=1}^{n} p(w_i \mid w_1 \cdots\cdots w_{i-1})$$

输入： 句子 S

输出： 句子概率p(S)

参数： $p(w_i|w_1,...,w_{i-1})$

3.1 语言模型基本概念

说明:

(1) w_i 可以是字、词、短语或词类等等，称为统计基元。通常以“词”代之。

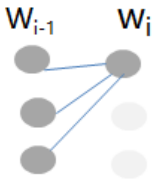
(2) w_i 的概率由 $w_1, \ldots, w_{i-1}$ 决定，由特定的一组$w_1, \ldots, w_{i-1}$ 构成的一个序列，称为 w_i 的历史（history）。

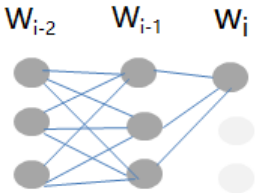
3.1 语言模型基本概念

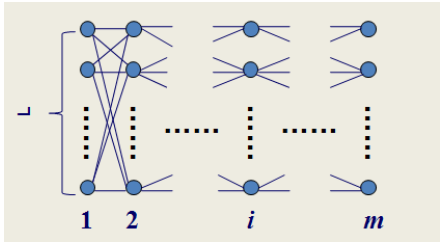
原始定义存在的问题：

对于$p(w_i|w_1,\ldots,w_{i-1})$：第 i（i >1）个统计基元，基元历史的个数为 i-1，如基元(如词汇表)有 L 个，i 基元就有 L^{i-1} 种不同的历史情况，模型有L^i个自由参数

如：$p(w_i|w_{i-1})$ 参数 3　　$p(w_i|w_{i-2}w_{i-1})$ 参数 9　　$p(w_m|w_1\ldots w_{m-1})$ 模型参数 L^m







如果 L=5000, m = 3, 自由参数的数目为 1250 亿!

一个汉语句子平均有22个词

问题解决方法

减少历史基元的个数，马尔可夫方法：假设任意一个词 w_i出现的概率只与它前面的w_{i-1} 有关，问题得以简化

$$p(S)=p(w_1)p(w_2|w_1)\ldots p(w_n|w_1,\ldots,w_{n-1})$$

$$p(s) = p(w_1)\times p(w_2|w_1)\times p(w_3|w_2)\times \ldots \times p(w_n|w_{n-1})$$

二元模型

3.1 语言模型基本概念

n 元文法(n-gram)

n-gram 模型假设一个词的出现概率只与它前面的*n*-1个词相关，距离大于等于*n*的上文词会被忽略

$$p(w_i|w_1,\ldots,w_{i-1}) \approx p(w_i|w_{i-(n-1)} \ldots,w_{i-1})$$

- 1 元文法模型 (unigram) : $p(w_1,\ldots,w_m)=\prod_{i=1}^{m} p(w_i)$ w_i 独立于历史
- 2 元文法模型（bigram): $p(w_1,\ldots,w_m)=\prod_{i=1}^{m} p(w_i|w_{i-1})$ w_i 保留前1个词序
- 3 元文法模型（trigram ）:$p(w_1,\ldots,w_m)=\prod_{i=1}^{m} p(wi|w_{i-2}, w_{i-1})$ w_i 保留前2个词序
- ……
- n 元文法模型（n-gram）:$p(w_1,\ldots,w_m)=\prod_{i=1}^{m} p(wi|w_{i-(n-1)} \ldots w_{i-1})$ w_i 保留前n个词序

n-gram 就是对 $p(w_i|w_1,\ldots,w_{i-1})$的简化程度而定义

3.1 语言模型基本概念

◆ 理论上讲，N 越大越好；但 N 越大，需要估计的参数越多

◆ 经验值：

tri-gram（3-gram）用的最多；

four-gram（4-gram）以上需要太多的参数，少用。

高阶模型也无法覆盖所有的语言现象

例： 给定句子：John read a book 求 概率

解： 增加标记：<BOS> John read a book <EOS>

基于1元文法的概率为：

$$p(\text{John read a book}) = p(\text{John}) \times p(\text{read}) \times p(\text{a}) \times P(\text{book})$$

基于2元文法的概率为：

$$p(\text{John read a book}) = p(\text{John}|\text{<BOS>}) \times p(\text{read}|\text{John}) \times p(\text{a}|\text{read}) \times p(\text{book}|\text{a}) \times p(\text{<EOS>}|\text{book})$$

问题：如何获得 n 元语法模型中的各概率值(参数)?

内 容 提 要

3.2.1 参数估计

参数估计（模型训练）：获得模型中所有的条件概率（模型参数）

1. 训练语料：

- 已知语料
- 训练语料应尽量和应用领域一致
- 语料尽量足够大
- 训练前应预处理

语言模型对于训练文本的类型、主题和风格等都十分敏感

3.2.1 参数估计

2. 参数学习的方法

对于 *n*-gram，参数 $p(w_i \mid w_{i-n+1}^{i-1})$ 可由最大似然估计求得:

$$p(w_i \mid w_{i-n+1}^{i-1}) = f(w_i \mid w_{i-n+1}^{i-1}) = \frac{\sum_{w_i} c(w_{i-n+1}^{i})}{\sum_{w_i} c(w_{i-n+1}^{i-1})}$$

其中:

$\sum_{w_i} c(w_{i-n+1}^{i-1})$ 是历史串 w_{i-n+1}^{i-1} 在给定语料中出现的次数

$\sum_{w_i} c(w_{i-n+1}^{i})$, 为 w_{i-n+1}^{i-1} 与 w_i 同现的次数。

最大似然估计(maximum likelihood Evaluation, MLE)

3.2.1 参数估计

例如： 给定训练语料：

“John read Moby Dick”

“Mary read a different book”

“She read a book by Cher”

如何 训练 2 元文法

$$p(\text{John read a book}) = p(\text{John}|\langle\text{BOS}\rangle) \times p(\text{read}|\text{John}) \times p(\text{a}|\text{read}) \times p(\text{book}|\text{a}) \times p(\langle\text{EOS}\rangle|\text{book})$$ 参数

3.2.1 参数估计

解： 参数：p(John|<BOS>)，p(read|John)，p(a|read)，p(book|a)，p(<EOS>|book)

$$p(John \mid <BOS>) = \frac{c(<BOS> John)}{\sum_w c(<BOS> w)} = \frac{1}{3} \qquad p(read \mid John) = \frac{c(John \quad read)}{\sum_w c(John \quad w)} = \frac{1}{1}$$

$$p(a \mid read) = \frac{c(read \quad a)}{\sum_w c(read \quad w)} = \frac{2}{3} \qquad p(book \mid a) = \frac{c(a \quad book)}{\sum_w c(a \quad w)} = \frac{1}{2}$$

$$p(<EOS> \mid book) = \frac{c(book \quad <EOS>)}{\sum_w c(book \quad w)} = \frac{1}{2}$$

语料：

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

3.2.1 参数估计

句子 *John read a book* 的概率

基于2元文法的概率为:

p(John read a book) = p(John|<BOS>)×p(read|John)×p(a|read)×

p(book|a)×p(<EOS>|book)

$$P\ (John\ read\ a\ book) = \frac{1}{3}\times 1\times \frac{2}{3}\times \frac{1}{2}\times \frac{1}{2}\approx 0.06$$

3.2.1 参数估计

问题： 如求，$p(Cher\ read\ a\ book) = ?$

$$=p(Cher|<BOS>)\times p(read|Cher)\times p(a|read)\times$$
$$p(book|a)\times p(<EOS>|book)$$

$$p(Cher\ |<\ BOS\ >)=\frac{c(<\ BOS\ >\ Cher)}{\sum_w c(<\ BOS\ >\ w)}=\frac{0}{3}$$

$$p(read\ |\ Cher)=\frac{c(Cher\quad read)}{\sum_w c(Cher\quad w)}=\frac{0}{1}$$

于是，$p(Cher\ read\ a\ book) = 0$ ❓

数据匮乏(稀疏) (*Sparse Data*) 引起零概率问题

$$p(read\ |\ John)=\frac{c(John\quad read)}{\sum_w c(John\quad w)}=\frac{1}{1}$$ ❓

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

内 容 提 要

3.2.2 数据平滑

1. 数据平滑的基本思想：

调整最大似然估计的概率值,使零概率增值，使非零概率下调，“劫富济贫”，消除零概率，改进模型的整体正确率。

基本目标：测试样本的语言模型困惑度越小越好。

基本约束：$\sum_{w_i} p(w_i \mid w_1, w_2, \cdots, w_{i-1}) = 1$

3.2.2 数据平滑

2. 数据平滑方法：

- 加1法(Additive smoothing)
- 减值法/折扣法 (Discounting)
  - 1) Good-Turing
  - 2) Back-off (Katz)
  - 3) 绝对减值(H. Ney)
  - 4) 线性减值
- 删除减值法：低阶代替高阶

3.2.2 数据平滑

◆ **加1法 (Additive smoothing)**

基本思想: 每一种情况出现的次数加1。

如，对于2-gram 有：

$$p(w_i \mid w_{i-1}) = \frac{1 + c(w_{i-1}w_i)}{\sum_{w_i}[1 + c(w_{i-1}w_i)]} = \frac{1 + c(w_{i-1}w_i)}{|V| + \sum_{w_i} c(w_{i-1}w_i)}$$

其中，V 为被考虑语料的词汇量（全部可能的基元数)

3.2.2 数据平滑

问题回顾：　　如求，$p(Cher\ read\ a\ book) = ?$

$$=p(Cher|<BOS>)\times p(read|Cher)\times p(a|read)\times p(book|a)\times p(<EOS>|book)$$

$$p(Cher|<BOS>)=\frac{c(<BOS>\ Cher)}{\sum_w c(<BOS>\ w)}=\frac{0}{3}$$

$$p(read|Cher)=\frac{c(Cher\quad read)}{\sum_w c(Cher\quad w)}=\frac{0}{1}$$

$$p(Cher\ read\ a\ book)=0$$

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

3.2.2 数据平滑

平滑处理：

原来：

$p(Cher|<BOS>) = 0/3$

$p(read|Cher) = 0/1$

$p(a|read) = 2/3$

$p(book|a) = 1/2$

$p(<EOS>|book) = 1/2$

平滑以后：

$p(Cher|<BOS>) = (0+1)/(11+3) = 1/14$

$p(read|Cher) = (0+1)/(11+1) = 1/12$

$p(a|read) = (1+2)/(11+3) = 3/14$

$p(book|a) = (1+1)/(11+2) = 2/13$

$p(<EOS>|book) = (1+1)/(11+2) = 2/13$

词汇量：$|V| = 11$

$p(Cher\ read\ a\ book)$

$= p(Cher|<BOS>) \times p(read|Cher) \times p(a|read) \times p(book|a) \times p(<EOS>|book)$

$$= \frac{1}{14} \times \frac{1}{12} \times \frac{3}{14} \times \frac{2}{13} \times \frac{2}{13} \approx 0.00003$$

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

3.2.2 数据平滑

对于句子John read a book 也需数据数据平滑处理：

原来：

$p(John|<BOS>) = 1/3,$

$p(read|John) = 1/1,$

$p(a|read) = 2/3,$

$p(book|a) = 1/2,$

$p(<EOS>|book) = 1/2$

平滑后：

$p(John|<BOS>) = 2/14,$

$p(read|John) = 2/12,$

$p(a|read) = 3/14,$

$p(book|a) = 2/13,$

$p(<EOS>|book) = 2/13$

$p(John\ read\ a\ book)$

$= p(John|<BOS>) \times p(read|John) \times p(a|read) \times p(book|a) \times p(<EOS>|book)$

$$= \frac{2}{14} \times \frac{2}{12} \times \frac{3}{14} \times \frac{2}{13} \times \frac{2}{13} \approx 0.0001$$

<BOS>John read Moby Dick<EOS>

<BOS>Mary read a different book<EOS>

<BOS>She read a book by Cher<EOS>

3.2.2 数据平滑

- **减值法/折扣法 (Discounting)**

基本思想：修改训练样本中事件的实际计数，使样本中(实际出现的)不同事件的概率之和小于1，剩余的概率量分配给未见概率。

1) Good-Turing　　2) Back-off (Katz)

3) 绝对减值(H. Ney)　　4) 线性减值

- **删除插值法(Deleted interpolation)**

基本思想：用低阶语法估计高阶语法，即当 3-gram 的值不能从训练数据中准确估计时，用 2-gram 来替代，同样，当 2-gram 的值不能从训练语料中准确估计时，可以用 1-gram 的值来代替。插值公式：

$$p(w_3 \mid w_1 w_2) = \lambda_3 p'(w_3 \mid w_1 w_2) + \lambda_2 p'(w_3 \mid w_2) + \lambda_1 p'(w_3)$$

其中，$\lambda_1 + \lambda_2 + \lambda_3 = 1$

（略）

3.2.2 数据平滑

神经网络语言模型不需要数据平滑　　**Why ?**

内 容 提 要

3.1　语言模型基本概念

3.2　语言模型参数估计

3.3　语言模型性能评价

3.4　语言模型应用

3.3 语言模型性能评价

目前主要有两种评价方法：

1. 实用方法：

通过查看该模型在实际应用（如拼写检查、机器翻译）中的表现来评价，优点是直观、实用，缺点是缺乏针对性、不够客观。

2. 理论方法：

用模型的 迷惑度/困惑度/混乱度（preplexity）衡量。其基本思想是能给测试集赋予较高概率值（低困惑度）的语言模型较好

3.3 语言模型性能评价

困惑度定义

平滑的 n-gram 模型句子的概率: $p(s)=\prod_{i=1}^{m+1} p(w_i \mid w_{i-n+1}^{i-1})$

假定测试语料 T 由 l_T 个句子构成 $(t_1, \ldots, t_{lT})$

则整个测试集的概率为: $p(T)=\prod_{i=1}^{l_T} p(t_i)$

模型 $p(w_i \mid w_{i-n+1}^{i-1})$ 对于测试语料的交叉熵: $H_p(T)=-\frac{1}{W_T}\log_2 p(T)$

其中，W_T 是测试文本 T 的词数。

模型 p 的**困惑度** $PP_p(T)$ 定义为: $PP_P(T)=2^{H_p(T)}$

内 容 提 要

3.1　语言模型基本概念

3.2　语言模型参数估计

3.3　语言模型性能评价

3.4　语言模型应用

3.4 语言模型应用

语言模型的用途

- 决定哪一个词序列的可能性更大
- 已知若干个词，预测下一个词

....

3.4 语言模型应用

例 1： 判断下列哪个句子更合理?

1. 美联储主席本．伯南克昨天告诉媒体 7000 亿美元的救助资金将给上百家银行、保险公司和汽车公司。 句概率≈ 10^{-20}

2. 本．伯南克美联储主席昨天7000 亿美元的救助资金告诉媒体将借给银行、保险公司和汽车公司上百家。 句概率≈ 10^{-25}

3. 联主美储席本．伯诉体南将借天的救克告媒昨助资金70元亿00 美给上百百百家银保行、汽车险公司公司和。 句概率≈ 10^{-70}

解：按 n-gram 模型计算：

结论： 第一个句子最有可能

3.4 语言模型应用

例2： 给定拼音串对应的汉字串？

给定拼音串：ta shi yan jiu sheng wu de

解：可能的汉字串：

{踏实研究生物的, 他实验救生物的，他是研究生物的，他使烟酒生雾的,...}

求每种可能汉字串的概率

如使用 2-gram：

$p(CString_1) = p(踏实|<BOS>) \times p(研究|踏实) \times p(生物|研究) \times p(的|生物) \times p(<EOS>|的)$

$p(CString_2) = p(他|<BOS>) \times p(实验|他) \times p(救|实验) \times p(生物|救) \times p(的|生物) \times p(<EOS>|的)$

……

结论：选择概率最大的字串

3.4 语言模型应用

例3： 已知若干个词，预测下一个词

用于联想输入法

如，基于 n-gram 的智能狂拼、微软拼音输入法等

语言模型变种：

◆ 前向-后向语言模型

$$\text{sentence}(t_1, t_2, \ldots, t_N)$$

前向语言模型：

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_1, t_2, \ldots, t_{k-1}).$$

后向语言模型：

$$p(t_1, t_2, \ldots, t_N) = \prod_{k=1}^{N} p(t_k \mid t_{k+1}, t_{k+2}, \ldots, t_N).$$

◆ K-Skipping N-gram Model

一个词的出现概率只与它前(后)面的距离为K的 n-1个词相关。核心思想是刻画远距离约束关系

◆ Class-based N-gram Mode

该方法基于词类建立语言模型，以缓解数据稀疏问题，且可以方便融合部分语法信息

如，n-pos 模型

语言模型变种：

- **指数语言模型**

  传统的n-gram语言模型，只是考虑了词形方面的特征，而没有词性以及语义层面上的知识最大熵模型MaxEnt、最大熵马尔科夫模型MEMM、条件随机域模型CRF可以更好的融入多种知识源，刻画语言序列特点，较好的用于解决序列标注问题。

- **Topic-based N-gram Mode**

  该方法将训练集按主题划分成多个子集，并对每个子集分别建立N-gram语言模型，以解决语言模型的主题自适应问题。

- **Cache-based N-gram Model**

  该方法利用cache缓存前一时刻的信息，以用于计算当前时刻概率，以解决语言模型动态自适应问题。

  ……

  详情请参考相关资料

参考文献：

宗成庆，统计自然语言处理（第2版）课件

吴军，数学之美，人民邮电出版社

在 此 表 示 感 谢!

谢谢各位！

Q&A