自然语言处理

杨海钦

2025-2026-1学期

基于Yulia, Diyi, Junjie等课件更新

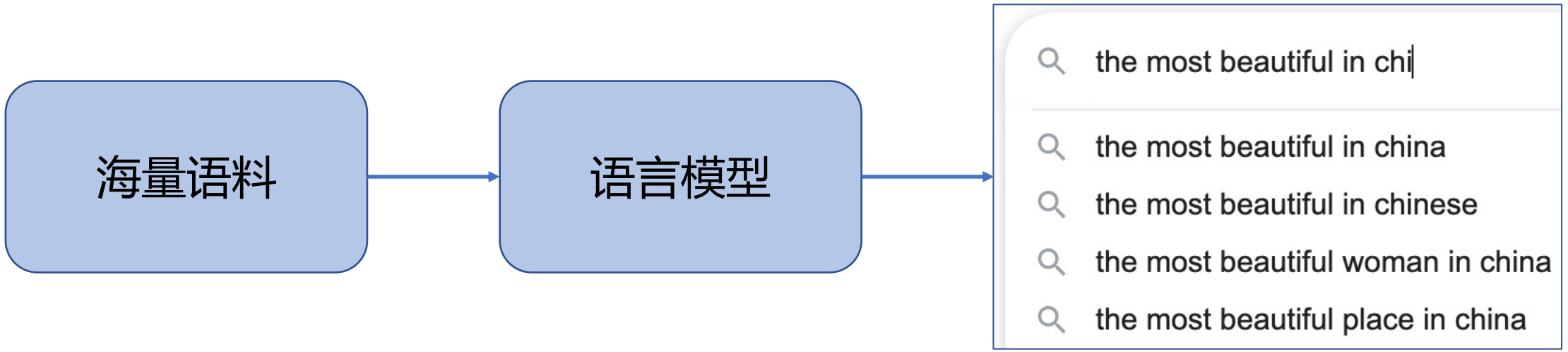
面试问题

- 什么是马尔可夫假设？为什么它在 N 元语言模型中是必需的？
- N 元语言模型面临 “数据稀疏问题”（部分 N-gram 未出现），请说明其危害，并列举 2 种核心解决方法及原理。
- 什么是困惑度（Perplexity）？它为何能作为语言模型的核心评估指标？

大纲

- 语言模型/Language modeling
  - 定义
  - 作用
  - N元语言模型
    - 马尔科夫假设
- 实际考量问题
- 语言模型的评估
  - 外部评估
  - 内在评估

语言模型/Language Modelling



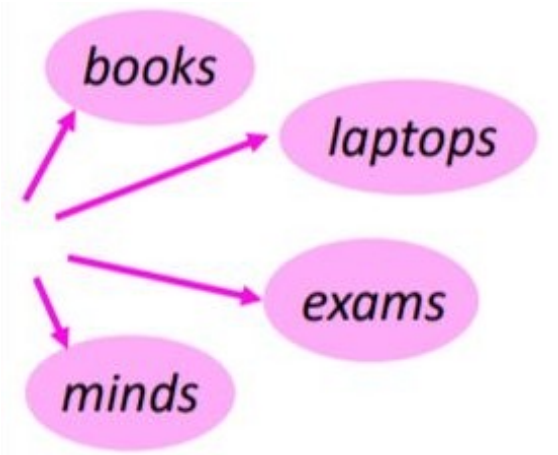
语言模型的作用

- 语法官/a judge of grammaticality
- 语义合理性的鉴定者/a judge of semantic plausibility
- 风格一致性的执行者/an enforcer of stylistic consistency
- 知识库/a repository of knowledge

语言模型任务定义

- 任务: 预测下一个单词是什么

the students opened their _____



- 定义: 给定词语序列: $\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(t)}$, 计算下一个词$\boldsymbol{x}^{(t+1)}$的概率分布，其中$\boldsymbol{x}^{(t+1)}$是字典$V = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_{|V|}\}$里的任意词

语言模型无处不在

- 输入法

- 搜索引擎

大语言模型是语言模型

自然语言处理的不足包括什么

自然语言处理（NLP）作为人工智能领域的核心方向之一，尽管在过去十年（尤其大语言模型爆发后）取得了突破性进展，但在**理论基础、技术落地、实际应用**等多个层面仍存在显著不足。以下从 **核心技术局限、实际应用挑战、伦理与安全问题** 三个维度，结合具体案例和研究现状，系统梳理 NLP 的主要不足（中英双语对照，便于教学和研究使用）：

一、核心技术局限（Core Technical Limitations）

1. 缺乏真正的语义理解（Lack of True Semantic Understanding）

语言模型

- 给定文本$\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(T)}$，该文本的概率(根据语言模型)是:

$$P\left(\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(T)}\right)$$
$$= P\left(\boldsymbol{x}^{(1)}\right)\times P\left(\boldsymbol{x}^{(2)}\middle|\boldsymbol{x}^{(1)}\right)\times\cdots\times P\left(\boldsymbol{x}^{(T)}\middle|\boldsymbol{x}^{(T-1)},\ldots,\boldsymbol{x}^{(1)}\right)$$
$$= P\left(\boldsymbol{x}^{(1)}\right)\prod_{t=2}^{T} P\left(\boldsymbol{x}^{(t)}\middle|\boldsymbol{x}^{(t-1)},\ldots,\boldsymbol{x}^{(1)}\right)$$

由语言模型提供

N元语言模型

“I have a dog whose name is Lucy. I have two cats, they like playing with Lucy.”

- 问题: 如何学习语言模型？
- 回答(前深度学习): 学习n元语言模型

N元语言模型

“I have a dog whose name is Lucy. I have two cats, they like playing with Lucy.”

- 定义: n元是n个连续单词/词元(token)组成的块
- 例子
  - 一元词组: {I, have, a, dog, whose, name, is, Lucy, two, cats, they, like, playing, with}
  - 二元词组: {I have, have a, a dog, dog whose, … , with Lucy}
  - 三元词组: {I have a, have a dog, a dog whose, … , playing with Lucy}
  - 四元词组: {I have a dog, … , like playing with Lucy}
  - …

N元语言模型

“I have a dog whose name is Lucy. I have two cats, they like playing with Lucy.”

- $\boldsymbol{w_1}$: 一元词
- $\boldsymbol{w_1 w_2}$: 二元词组
- $\boldsymbol{w_1 w_2 w_3}$: 三元词组
- $\boldsymbol{w_1 w_2 \cdots w_n}$: n元词组

N元语言模型

“I have a dog whose name is Lucy. I have two cats, they like playing with Lucy.”

- 问题: 如何学习语言模型？
- 回答(前深度学习): 学习n元语言模型

- 思路: 统计不同n元词组的频率，并用这些数据预测下一个单词

一元词组概率unigram probability

“I have a dog whose name is Lucy. I have two cats, they like playing with Lucy.”

- 语料词数: $m = 17$
- $P(\text{Lucy}) = \frac{2}{17}$; $P(\text{cats}) = \frac{1}{17}$

- Unigram probability: $P(w) = \frac{count(w)}{m} = \frac{C(w)}{m}$

二元词组概率bigram probability

"I have a dog whose name is Lucy. I have two cats, they like playing with Lucy."

- $P(A|B) = \frac{P(A,B)}{P(B)}$
- $P(\text{have} \mid \text{I}) = \frac{C(\text{I have})}{C(\text{I})} = \frac{2}{2} = 1$
- $P(\text{two} \mid \text{have}) = \frac{C(\text{have two})}{C(\text{have})} = \frac{1}{2} = 0.5$
- $P(\text{eating} \mid \text{have}) = \frac{C(\text{have eating})}{C(\text{have})} = \frac{0}{2} = 0$

- Bigram prob.:
  - $P(w_2|w_1) = \frac{C(w_1 w_2)}{\sum_w C(w_1 w)} = \frac{C(w_1 w_2)}{C(w_1)}$

三元词组概率trigram probability

"I have a dog whose name is Lucy. I have two cats, they like playing with Lucy."

- $P(A|B) = \frac{P(A,B)}{P(B)}$
- $P(\text{a} \mid \text{I have}) = \frac{C(\text{I have a})}{C(\text{I have})} = \frac{1}{2} = 0.5$
- $P(\text{several} \mid \text{I have}) = \frac{C(\text{I have several})}{C(\text{I have})} = \frac{0}{2} = 0$

- Trigram prob.:
  - $P(w_3|w_1w_2) = \frac{C(w_1w_2\ w_3)}{\sum_w C(w_1\ w_2\ w)} = \frac{C(w_1w_2\ w_3)}{C(w_1w_2)}$

N元词组概率
N-gram probability

“I have a dog whose name is Lucy. I have two cats, they like playing with Lucy.”

- $P(A|B) = \frac{P(A,B)}{P(B)}$
- N-gram prob.:
  - $P(w_n|w_1 w_2 \cdots w_{n-1}) = \frac{C(w_1 w_2 \ \cdots w_{n-1}\, w_n)}{C(w_1\, w_2 \cdots \ w_{n-1})}$

句子/段落/书本概率

$$P\left(\boldsymbol{x}^{(1)},\ldots,\boldsymbol{x}^{(T)}\right)=P\left(\boldsymbol{x}^{(1)}\right)\times P\left(\boldsymbol{x}^{(2)}\middle|\boldsymbol{x}^{(1)}\right)\times\cdots\times P\left(\boldsymbol{x}^{(T)}\middle|\boldsymbol{x}^{(T-1)},\ldots,\boldsymbol{x}^{(1)}\right)$$
$$=P\left(\boldsymbol{x}^{(1)}\right)\prod_{t=2}^{T}P\left(\boldsymbol{x}^{(t)}\middle|\boldsymbol{x}^{(t-1)},\ldots,\boldsymbol{x}^{(1)}\right)$$

P(I have a dog whose name is Lucy) =
P(I) ×
P(have | I) ×
P(a | I have) ×
… ×
P(Lucy | I have a dog whose name is)

如何估算？

马尔可夫假设/Markov Assumption


Andrei Markov

- 马尔科夫假设: $\boldsymbol{x}^{(t+1)}$只依赖于前面的$n-1$个单词
  - 马尔可夫链: *一种描述一系列可能事件的随机模型，其中每个事件的概率仅取决于前一个事件所达到的状态*。

$$P\left(\boldsymbol{x}^{(t+1)}\middle|\boldsymbol{x}^{(t)},\ldots,\boldsymbol{x}^{(1)}\right)=\underbrace{P\left(\boldsymbol{x}^{(t+1)}\middle|\boldsymbol{x}^{(t)},\ldots,\boldsymbol{x}^{(t-n+2)}\right)}_{n-1\text{ 个词}}$$

P(Lucy | I have a dog whose name is) $\cong P$(Lucy | name is)

或 $\cong P$(Lucy | is)

马尔可夫过程

- 链式法则

$$P(X_1 = x_1, X_2 = x_2, \cdots, X_n = x_n)$$

$$= P(X_1 = x_1) \prod_{t=2}^{n} P(X_i = x_i | X_1 = x_1, \cdots, X_{i-1} = x_{i-1})$$

- 一阶假设

$$= P(X_1 = x_1) \prod_{i=2}^{n} P(X_i = x_i | X_{i-1} = x_{i-1})$$

- 二阶假设

$$= P(X_1 = x_1) \times P(X_2 = x_2 | X_1 = x_1) \prod_{i=3}^{n} P(X_i = x_i | X_{i-1} = x_{i-1}, X_{i-2} = x_{i-2})$$

三元语言模型/Trigram LM

- 三元语言模型包含
  - 词典V
  - 对每个三元组，存在非负参数$q(w|u,v)$，使得

$$w \in V \cup \{[\text{EOS}]\}, u, v \in V \cup \{[\text{BOS}]\}$$

  - 句子 $x_1 x_2 \cdots x_n$（其中$x_n = [\text{EOS}]$）的概率是

$$P(x_1, \cdots, x_n) = \prod_{i=1}^{n} q(x_i | x_{i-1}, x_{i-2})$$

    - 其中 $x_0 = x_{-1} = [\text{BOS}]$

In-class Practice

大纲

- 语言模型/Language modeling
  - 定义
  - 作用
- N元语言模型
  - 一元组
  - 二元组
  - N元组
  - 马尔科夫假设
- 实际考量问题
- 语言模型的评估
  - 外部评估
  - 内在评估

实际考量

- 将非常小的数字相乘会产生数值下溢/underflow
  - **方案**: 在对数空间中做所有的运算
  - **优势**: 加法也比乘法快

马尔可夫假设是错的
Markovian Assumption is False

"He is from France, so it makes sense that his first language is ..."

- 需要对更长的依赖关系进行建模

稀疏性

- 估计概率q的极大似然
  - 设$C(w_1 w_2 \cdots w_{n-1}\, w_n)$为n元组(n-gram)在语料库中出现的次数

$$q(w_i|w_{i-2}\, w_{i-1}) = \frac{C(w_{i-2}\, w_{i-1}\ w_i)}{C(w_{i-2}\, w_{i-1})}$$

- 词汇量为20000个单词时，参数个数:

$$\Rightarrow 8 \times 10^{12}\,!$$

Bias-Variance Tradeoff

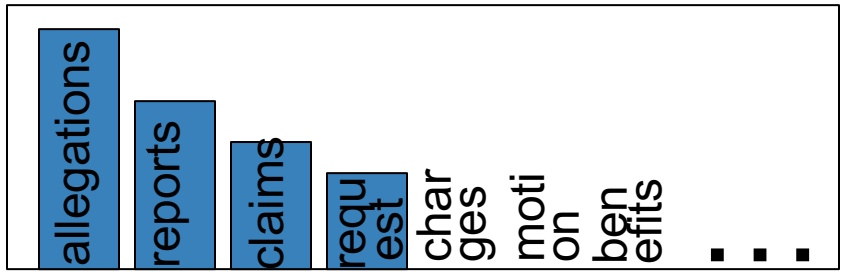
- 给定长度为M的语料
  - 三元组模型:
    - $q(w_i|w_{i-2}\ w_{i-1}) = \frac{C(w_{i-2}\ w_{i-1}\ \ w_i)}{C(w_{i-2}, w_{i-1})}$
  - 二元组模型:
    - $q(w_i|w_{i-1}) = \frac{C(\ w_{i-1}\ \ w_{si})}{C(w_{i-1})}$
  - 一元组模型
    - $q(w_i) = \frac{C(w_i)}{M}$

处理稀疏性

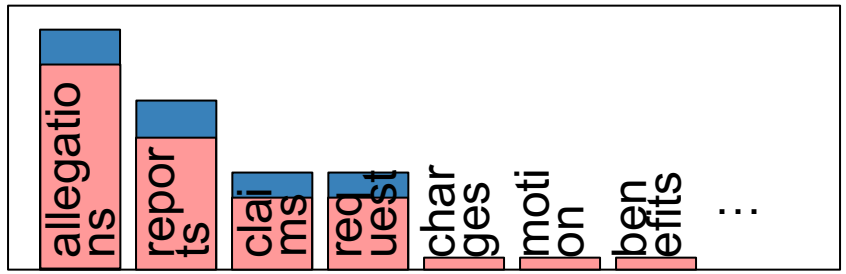
- 对于大多数n元组，我们只有少量的样本
- 一般方法：修改观察到的计数改进评估方法
  - 回退/Backoff:
    - 如果有很好的根据，则使用三元组模型
    - 否则是二元组，不然是一元组
  - 插值/Interpolation：使用相关密集历史的估计组合来估计n元组的近似计数
  - 贴现/Discounting或平滑/Smoothing：通过对已观察事件的贴现计数来分配未观察事件的概率质量
    - Laplace平滑

Discounting/Smoothing Methods

- 我们经常需要从稀疏统计中做出估计：
  - P(w | denied the)
    - 3 allegations
    - 2 reports
    - 1 claims
    - 1 request
    - 7 total



- 平滑使尖锐的/spiky分布变得平坦flatten，因此它们可以更好地泛化：



In-class Practice II: 加州伯克利餐厅项目(Laplacian Smoothing)

假设: $|V| = 1446$

线性插值/Linear Interpolation

- 结合这三种模式获得所有的好处

$$q_{LI}(w_i|w_{i-2}\ w_{i-1}) = \lambda_1 \times q(w_i|w_{i-2}\ w_{i-1}) + \lambda_2 \times q(w_i|\ w_{i-1}) + \lambda_3 \times q(w_i)$$

  - 其中 $\lambda_i \geq 0, \lambda_1 + \lambda_2 + \lambda_3 = 1$
- 注意: 需要验证参数定义一个概率分布

$$\sum_{w \in \mathcal{V}} q_{LI}(w|u\ v) = \sum_{w \in \mathcal{V}} \lambda_1 \times q(w|u\ v) + \lambda_2 \times q(w|v) + \lambda_3 \times q(w)$$

$$= \lambda_1 \sum_{w \in \mathcal{V}} q(w|u\ v) + \lambda_2 \sum_{w \in \mathcal{V}} q(w|v) + \lambda_3 \sum_{w \in \mathcal{V}} q(w)$$

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

处理未登记词/Out-of-Vocabulary Terms

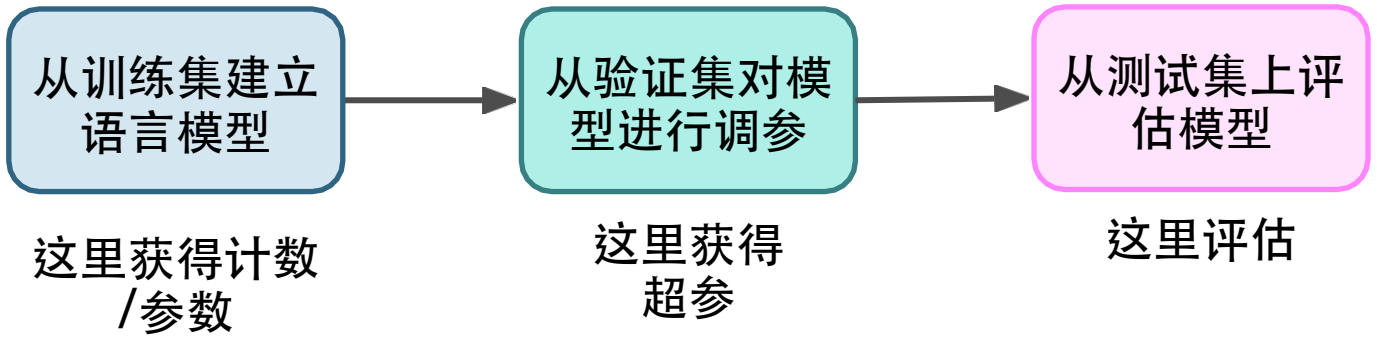
- 定义一个特殊的OOV或“未知”符号<unk>。将训练数据中的一些（或全部）生疏词转换为<unk>
  - 不能公平地比较应用不同处理方法的两种语言模型
- 在字符级别建立语言模型
- 其他方法: 如BPE

大纲

- 语言模型/Language modeling
  - 定义
  - 作用
- N元语言模型
  - 一元组
  - 二元组
  - N元组
  - 马尔科夫假设
- 实际考量问题
- 语言模型的评估
  - 外部评估
  - 内在评估

评估/Evaluation

- 直观地说，语言模型应该给它们从未见过的真实语言赋予高概率
  - 想要在保留集(held-out)而不是在训练数据上最大化其可能性，
- 来自计数/充分统计的模型需要在保留集上调整泛化参数以模拟测试集上的泛化程度
- 设置超参数以最大化保留数据的可能性（通常使用网格搜索grid search或EM)



评估/Evaluation

- **外部评估/Extrinsic evaluation**：建立一个新的语言模型，将其用于某些任务（MT， ASR等）
- **内在评估/Intrinsic evaluation**：衡量语言建模的能力

N元组语言模型的外部评估/Extrinsic evaluation

- 比较模型A和B的最佳评估
  - 将每个模型放入一个任务中
    - 如: 拼写校正/Spelling corrector，语音识别/speech recognition，机器翻译系统/MT system
  - 运行任务，得到A和B的准确率
    - 有多少拼写错误的单词被正确纠正了
    - 有多少单词被正确翻译
- 比较A和B的准确性

N元组语言模型的外在(内在)评估困难

- 外在的评价
  - 耗时: 可能需要几天或几周

- 所以
  - 有时用内在评价：困惑度/perplexity
    - 坏的近似
      - 除非测试数据和训练数据分布相近
    - 所以通常只在论文实验中有用

内在评估: 困惑度Perplexity

- 测试集: $\mathcal{S} = \{s_1, s_2, \cdots, s_T\}$
  - 参数由训练数据估计得到

$$p(\mathcal{S}) = \prod_{i=1}^{T} p(s_i)$$

$$\log_2 p(\mathcal{S}) = \sum_{i=1}^{T} \log_2 p(s_i)$$

$$\begin{aligned} &P([\text{BOS}]\ \text{I have a dog}\ [\text{EOS}]) \\ &= q(\text{I} \mid [\text{BOS}], [\text{BOS}]) \times \\ &\quad q(\text{have} \mid [\text{BOS}], \text{I}) \times \\ &\quad q(\text{a} \mid \text{I have}) \times \\ &\quad q(\text{dog} \mid \text{have a}) \times \\ &\quad q([\text{EOS}] \mid \text{a dog}) \end{aligned}$$

  - T: 测试数据句子的数量

内在评估: 困惑度Perplexity

- 测试集: $S = \{s_1, s_2, \cdots, s_T\}$
  - 参数由训练数据估计得到

$$p(S) = \prod_{i=1}^{T} p(s_i)$$

$$\log_2 p(S) = \sum_{i=1}^{T} \log_2 p(s_i)$$

$$\text{perplexity} = 2^{-l}, \qquad l = \frac{1}{M} \sum_{i=1}^{T} \log_2 p(s_i)$$

  - T: 测试数据句子的数量
  - M: 测试语料中词的数量
  - 好的语言模型
    - 较高的$p(S)$和较低的困惑度

理解困惑度

$$\text{perplexity} = 2^{-\frac{1}{M}\sum_{i=1}^{T}\log_2 p(s_i)}$$

- 这是一个语言的分支因子/branching factor
  - 赋予测试数据概率为1时:
    - $\Longrightarrow$ $\text{perplexity} = 1$
  - 赋予每个单词概率为$1/|V|$时:
    - $\Longrightarrow$ $\text{perplexity} = |V|$
  - 赋予任意单词概率为0时:
    - $\Longrightarrow$ $\text{perplexity} = \infty$
    - 这激发适当的概率约束

$$\sum_{\mathbf{e}\in\Sigma^*} p_{\text{LM}}(\mathbf{e}), \qquad \text{s.t. } p_{\text{LM}}(\mathbf{e}) \geq 0, \forall \mathbf{e} \in \Sigma^*$$

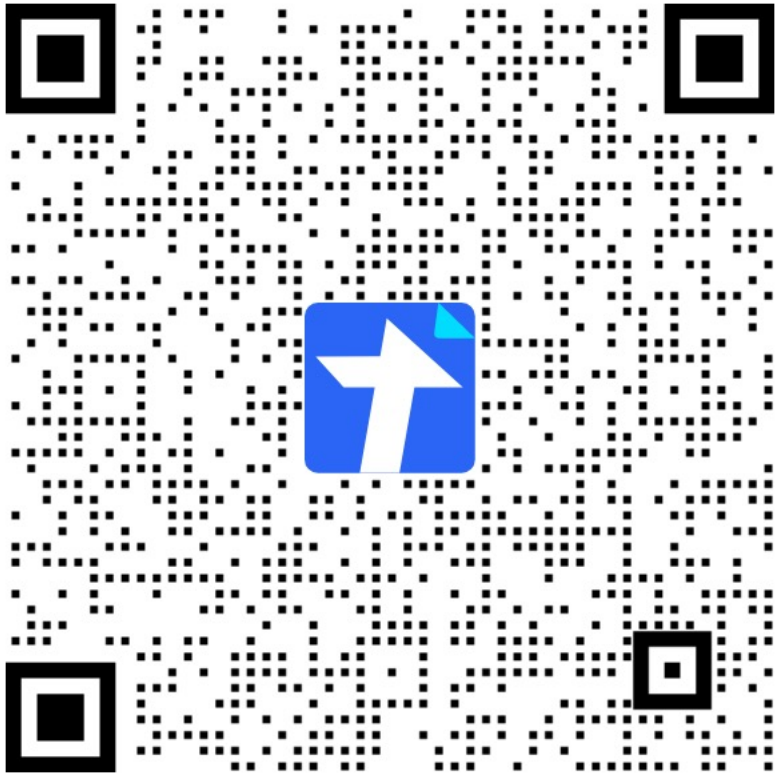
- 注意: 不能比较在不同语料库上训练的语言模型的困惑度

典型困惑度值

- 训练语料：《华尔街日报》新闻
  - 词汇量 $|V|$: 38,000,000
- 测试集: $|V| = 1{,}500{,}000$
- 困惑度
  - 一元组模型: 962
  - 二元组模型: 170
  - 三元组模型: 109 (<< 1,500,000)

一句话总结

- 语言模型
  - 定义、计算(马尔科夫假设)
- 计算语言模型的考虑因素
  - 长上下文
  - 稀疏性
  - OOV
- 语言模型的评估
  - 外在评估
  - 内在评估/Perplexity



- 课外阅读
  - [SLP3] Ch. 3

In-class Practice: 加州伯克利餐厅项目 Berkeley Restaurant Project

- 该项目是目前不存在/上个世纪的对话系统，可以回答关于加州伯克利餐馆相关的问题
- 句子范例
  - can you tell me about any good cantonese restaurants close by
  - mid priced that food is what i’m looking for
  - tell me about chez pansies
  - can you give me a listing of the kinds of food that are available
  - i’m looking for a good place to eat breakfast
  - when is caffe venezia open during the day

In-class Practice: 加州伯克利餐厅项目原始二元组计算

	i	want	to	eat	chinese	food	lunch	spend
i	5	827	0	9	0	0	0	2
want	2	0	608	1	6	6	5	1
to	2	0	4	686	2	0	6	211
eat	0	0	2	0	16	2	42	0
chinese	1	0	0	0	0	82	1	0
food	15	0	15	0	1	4	0	0
lunch	2	0	0	0	0	1	0	0
spend	1	0	1	0	0	0	0	0

In-class Practice: 加州伯克利餐厅项目二元组概率

i	want	to	eat	chinese	food	lunch	spend
2533	927	2417	746	158	1093	341	278

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

In-class Practice: 加州伯克利餐厅项目

- 已知
  - $P(\text{I}|[\text{BOS}]) = \ 0.25, P(\text{english}|\text{want}) = 0.0011$
  - $P(\text{food}|\text{englis}h) = \ 0.5, P([\text{EOS}]|\text{food}) = 0.68$
- 求下面句子的生成概率(便于计算，仅假设一阶马尔科夫关系)
  - S1: I want English food
  - S2: I want to eat Chinese food

Ans. to In-class Practice: 加州伯克利餐厅项目

解答.

因为，S1: I want English food，我们需要计算

$P(\text{S1}) = P(\text{I}|[\text{BOS}]) * P(\text{want}|\text{I}) * P(\text{english}|\text{want})$
$* P(\text{food}|\text{english}) * P([\text{EOS}]|\text{food})$

- $P(\text{I}|[\text{BOS}]) = 0.25$
- $P(\text{want}|\text{I}) = \frac{c(\text{I want})}{c(\text{I})} = \frac{827}{2533} = 0.33$
- $P(\text{english}|\text{want}) = 0.0011$
- $P(\text{food}|\text{englis}h) = 0.5$
- $P([\text{EOS}]|\text{food}) = 0.68$

因此，$P(\text{S1}) = 0.25 * 0.33 * 0.0011 * 0.5 * 0.68 = 3.09\text{e}{-5}$

因为S2: I want to eat Chinese food，我们需要计算

$P(\text{S2})$
$= P(\text{I}|[\text{BOS}]) * P(\text{want}|\text{I}) * P(\text{to}|\text{want}) * P(\text{eat}|\text{to})$
$* P(\text{chinese}|\text{eat}) * P(\text{food}|\text{chinese})$
$* P([\text{EOS}]|\text{food})$

- $P(\text{I}|[\text{BOS}]) = 0.25$
- $P(\text{want}|\text{I}) = \frac{827}{2533} = 0.33$, $P(\text{to}|\text{want}) = \frac{608}{927} = 0.66$
- $P(\text{eat}|\text{to}) = 0.28$, $P(\text{chinese}|\text{eat}) = 0.021$
- $P(\text{food}|\text{chinese}) = 0.52$
- $P([\text{EOS}]|\text{food}) = 0.68$

因此，$P(\text{S2}) = 0.25 * 0.33 * 0.66 * 0.28 * 0.021 * 0.52 * 0.68 = 1.13\text{e}{-4}$