0. 写在前面

1 本课程总体结构

章节		教学内容
第一章 引言 (刘均, 2)		概念与研究背景；主要任务；挑战与研究方向；相关资源
第二章：自然语言的统计特性 (刘均, 1)		Zipf定律、Heaps定律、Benford 定律。
第三章：语言模型	词袋模型 (刘均, 3)	语言模型；词袋模型（BoW）；TF-IDF。 NLU任务：情感分析、文本聚类。
	概率语言模型 (李辰, 6)	概率语言模型；n-gram 模型；最大似然估计；平滑技术。 NLU任务：分词、语义关系抽取。
	主题模型 (刘均, 6)	生成模型；主题模型的图表示；LSA、PLSA、LDA；NMF等。 NLU任务：话题检测、推荐。
	神经网络语言模型 (李辰, 6)	分布式表示；C&W、CBOW、Skip-Gram、Glove等。 NLU任务：对话、实体消歧。
第四章：机器翻译	概述 (李辰, 1)	面临的挑战；发展历程；方法类别及特点；MT评估。
	统计机器翻译 (李辰, 3)	统计MT；Noisy Channel模型；IBM模型。
	神经网络机器翻译 与大语言模型 (刘均, 4)	RNN与LSTM简介；Encoder-Decoder框架；Attention模型；大语言模型。

- 这门课由于由两门老师授课，个人感觉结构比较混乱
- 由于时间紧任务重经费无，所以笔记还是按PPT内容和以上结构展开，即使有很多不合理的地方

2 考试有关事项

1. 关于语言模型的预备知识

1.1. 语言模型概念

1 含义：自然语言在不同语言单位上的数学模型→实现自然语言的可计算性

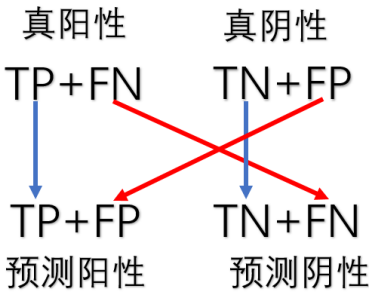
2 类型概览

模型	含义
词袋语言模型	用文中词汇表示文本
概率语言模型	根据给定词汇序列来预测下一个词汇的概念
主题语言模型	利用非监督方法获得文档中隐含的主题
神经网络语言模型	利用神经网络学习词汇/句子/字符

1.2. 语言模型的评价指标

1 召回率与精确度

1. 真假性&阴阳性：



实际\预测	$\mathbf{C}_1$	$\neg\mathbf{C}_1$
$\mathbf{C}_1$	True Positives(TP)真阳性	False Negatives(FN)假阴性
$\neg\mathbf{C}_1$	False Positives(FP)假阳性	True Negatives(TN)真阴性

2. 召回率与精确率

- $\text{Recall}=\frac{\text{TP}}{\text{TP}+\text{FN}}$：表示真实阳性中/被预测为阳性的比率
- $\text{Precision}=\frac{\text{TP}}{\text{TP}+\text{FP}}$：表示被预测为阳性中/真实阳性的比率，一词多义会使之降低
- $\text{F1-score}=2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$：二者的调和平均

3. 一词多义&一义多词

- 一词多义→多义间不相关的含义被认为相关→假阳性增加→Precision降低
- 一义多词→多词间相同的含义被认为是不同→假阴性增加→Recall降低

2 困惑度

1. 含义：反应不确定性，即困惑度越低→模型预测下一个元素时选择更少→预测越准确
2. 公式：$\text{Perplexit}=2^{\text{Cross-Entropy}}=\exp\left(-\frac{1}{N}\sum_{i=1}^{N}\ln P\left(w_i|w_1^{i-1}\right)\right)$
   - $P\left(w_i|w_1^{i-1}\right)$是谟型给定前序$w_1^{i-1}$条件下，预测词$w_i$的概率

1.3. 自然语言的统计特性

1.3.1. Zipf定律

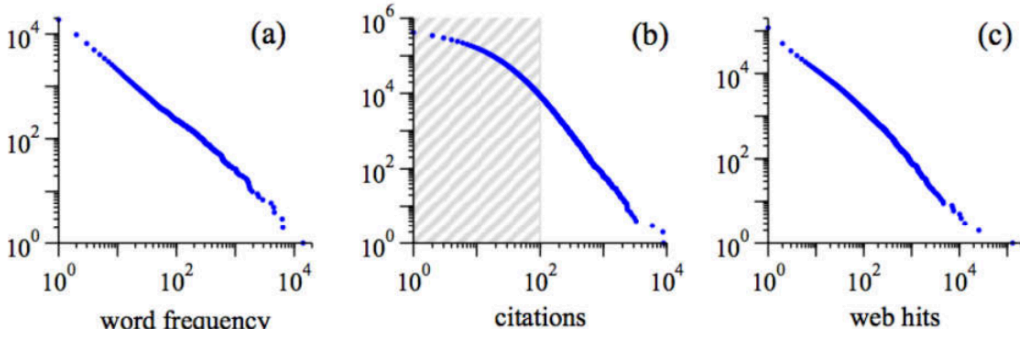
1 Zipf定律

1. 内容：令出现频率第r高的词汇出现频率为$f(r)$，则有$f(r)=\frac{\text{Const}}{r^s}$其中$s\approx 1$
2. 含义：对于词频分布，最常见词的分布极为普遍+大多数词出现频率极低
3. 解释：

解释模型	含义
米勒猴实验	胡乱生成的带有字母+空格的序列，词频和排名也符合幂律关系
最小努力原则	通过词频差异最小化交流的成本
优先连接机制	网络结构中，新节点倾向于连接度数更大的点，与Zipf类似

2 Zipf定律的实验

1. 符合程度：$f(r)=\frac{\text{Const}}{r^s} \rightarrow \log f(r)=\log C - s\log r$故可通过检测后者线性程度
2. 实验结论：幂律分布很常见+排名靠中间的术语会更符合



3 Zipf定律与索引

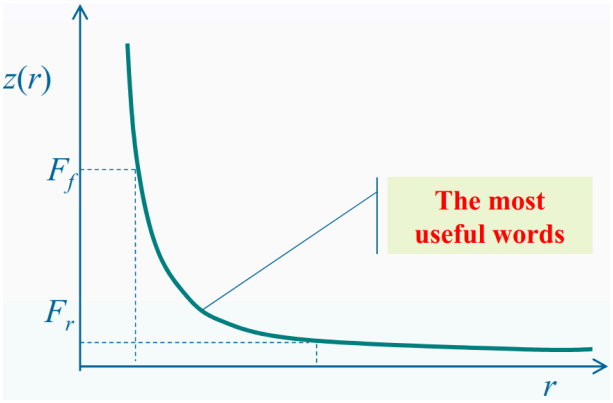
0. 倒排索引：用于快速全文检索的数据结构，示例如下
    - 文档

```
Doc1: fat cat rat rat
Doc2: fat cat
Doc3: fat
```

    - 构建的倒排索引

```
fat: Doc1 Doc2 Doc3
cat: Doc1 Doc2
rat: Doc1
```

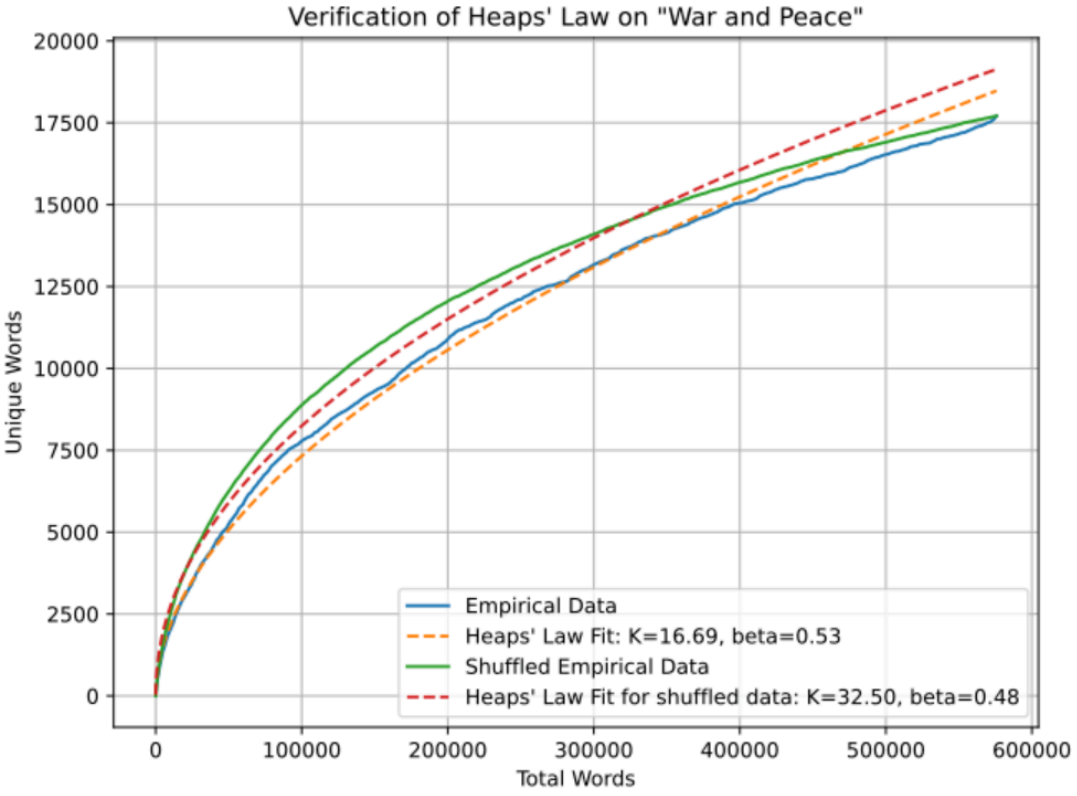
1. 词频太高/太低的词都不适合索引，会导致返回太多/太少的文档，适中的才最有价值



2. 基于Zipf定律，去处高频Stopword能优化倒排索引时空开销，如下为倒排索引的一个实例

1.3.2. Heaps定律

1 Heaps定律



1. 内容：词汇表大小V与文本词数n满足$V=Kn^{\beta}$
2. 参数：$10\leq K\leq 100$且$0.4\leq\beta\leq 0.6$，当$K=44$与$\beta=0.49$最匹配

2 用途：预测随文本增长词汇表&倒排索引大小的变化

1.3.1. Benford定律(第一数字法则)

1 Benford定律

1. 背景：在许多社会现象中，数据首位数往往分布不均(为1概率最大$\xrightarrow{\text{依次递减}}$为9概率最小)
2. 定律：令数据集中d作为首字母的概率$P(d)=\lg\left(1+\frac{1}{d}\right)$，$d>9$及非十进制时依旧适用

2 对Benford定律的一些思考

1. 适用：跨数量级变化的数据集，如财务数据和自然现象
2. 应用：检测数据造假、异常值、验证财务报告真实性
3. 成因：还不具备完全的可解释性，大概是因为数据在对数尺的分布

2. 词袋语言模型

2.1. BoW模型

1 基本步骤：以句I love machine learning以及Machine learning is fun为例

步骤	示例
分词	I \ love \ machine \ learning \ Machine \ learning \ is \ fun
词汇表	V=[I, love, machine,lerning, is, fun]
向量化	第一句变为A_1=[1,1,1,1,0,0]第二局变为A_2=[0,0,1,1,1,1]

2 特点

1. 原理上：完全忽略了语法/词序，默认词与词间的概率分布独立
2. 效果上：
    - 优点：实现极其简单，但高效且应用广泛
    - 缺点：无法区分&一义多词，如同义词替换后的两文档相似度低于实际值

2.2. TF-IDF模型

1 TF-IDF值

1. 计算:

$$\text{TF-IDF}(t,d)=\text{TF}(t,d)\times\text{IDF}(t)\rightarrow\begin{cases}\text{词频}\mathrm{TF}(t,d)=\dfrac{\text{词}t\text{在文档}d\text{出现次数}}{\text{文档}d\text{总词数}}\\ \text{逆文档频}\mathrm{IDF(t)}=\log\dfrac{\text{文档总数}}{\mathrm{DF}(t)(\text{包含}t\text{的文档数})+1}\end{cases}$$

2. 含义: $\text{TF-IDF}(t,d)$越高，代表词t对文档d越重要

2 TF-IDF值改进：原始词频值往往不是所需的

1. 对原始词频$\mathrm{TF}(t,d)$的改进

词频类型	公式	意义
对数	$1+\log(\mathrm{TF}(t,d))$	压缩较高词频，减少其对相关性影响的夸大
增强	$0.5+\dfrac{0.5\times\mathrm{TF}(t,d)}{\max_t\mathrm{TF}(t,d)}$	映射词频到$0.5\rightarrow1$，防止高频词权重过大
布尔	$\begin{cases}1 & \text{if } \mathrm{TF}(t,d)>0\\ 0 & \text{otherwise}\end{cases}$	不关注具体的词频值，仅表示是否出现
平均对数	$\dfrac{1+\log\left(\mathrm{TF}(t,d)\right)}{1+\log\left(\mathrm{ave}_{\mathrm{t}\in\mathrm{d}},(\mathrm{TF}(t,d))\right)}$	使词频高的词与低的词之间的差距不会过大

2. 对文档频率$\mathrm{DF}(t)$的改进：N是文档总数

文档频率$\mathrm{DF}(t)$	公式	意义
逆文档频率$\mathrm{IDF}(t)$	即$\log \frac{N}{\mathrm{DF}(t)}$者 $\log \frac{N}{\mathrm{DF}(t)+1}$	衡量词在文档集合中的稀有性
概率文档频率 $\mathrm{ProbDF}(t)$	$\max\left\{0, \log \frac{N-\mathrm{DF}(t)}{\mathrm{DF}(t)}\right\}$	通过概率角度评估词的稀有性

3. 归一化：对于

$$\mathbf{TF\text{-}IDF}=\begin{bmatrix} \text{TF-IDF}(t_1,d_1) & \text{TF-IDF}(t_1,d_2) & \cdots & \text{TF-IDF}(t_1,d_n) \\ \text{TF-IDF}(t_2,d_1) & \text{TF-IDF}(t_2,d_2) & \cdots & \text{TF-IDF}(t_2,d_n) \\ \vdots & \vdots & \ddots & \vdots \\ \text{TF-IDF}(t_m,d_1) & \text{TF-IDF}(t_m,d_2) & \cdots & \text{TF-IDF}(t_m,d_n) \end{bmatrix}$$

归一类型	公式	意义
余弦归一	$\mathbf{TF\text{-}IDF}\times\frac{1}{\sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}[\text{TF-IDF}(t_i,d_j)]^2}}$	用于计算文档间的余弦相似度
基准归一	$\mathbf{TF\text{-}IDF}\times\frac{1}{mn}$	消除文档集合大小对权重的影响
字长归一	$\mathbf{TF\text{-}IDF}\times\frac{1}{(\text{CharLen})^{\alpha}}$	适用于不同长度的文档，且$\alpha<1$

3 基于TF-IDF的余弦相似度

1. TF-IDF值：对于文档d_1和d_2，词汇表长为m

- $\mathbf{TF\text{-}IDF}=\begin{bmatrix} \text{TF-IDF}(t_1,d_1) & \text{TF-IDF}(t_1,d_2) \\ \text{TF-IDF}(t_2,d_1) & \text{TF-IDF}(t_2,d_2) \\ \vdots & \vdots \\ \text{TF-IDF}(t_m,d_1) & \text{TF-IDF}(t_m,d_2) \end{bmatrix} \xrightarrow{\text{余弦归一化}} \begin{bmatrix} \text{tf-idf}(t_1,d_1) & \text{tf-idf}(t_1,d_2) \\ \text{tf-idf}(t_2,d_1) & \text{tf-idf}(t_2,d_2) \\ \vdots & \vdots \\ \text{tf-idf}(t_m,d_1) & \text{tf-idf}(t_m,d_2) \end{bmatrix}$

2. 两文档余弦值：

- 未归一化表示：$\mathrm{sim}(d_1,d_2)=\frac{\sum_{j=1}^{m}\text{TF-IDF}(t_j,d_1)\cdot\text{TF-IDF}(t_j,d_2)}{\sqrt{\sum_{j=1}^{m}(\text{TF-IDF}(t_j,d_1))^2}\cdot\sqrt{\sum_{j=1}^{m}(\text{TF-IDF}(t_j,d_2))^2}}$
- 归一化表示：$\mathrm{sim}(d_1,d_2)=\sum_{j=1}^{m}\text{tf-idf}(t_j,d_1)\cdot\text{tf-idf}(t_j,d_2)$