基于fastText的文本分类应用
----情感计算

计算机学院 1150310609

王陈阳

2019.4.20

目录

- word2vec
- fastText
- 分类任务及数据分析
- 系统设计
- 实验结果
- Demo展示

word2vec

- 基本思想：采用一定的模型将词语映射到同一个坐标系，转换为数值型向量。
- 语言的特点：逻辑性（即词语的前后关系）
  - 原句：自然语言处理是计算机科学领域与人工智能领域中的一个重要方向。
  - 打乱后：方。自理要是计学领域机科域与向能个人领中的一工语然重智言处算
- 一个句子出现的概率：
  - 如果一个句子S由n个词w1~wn，那么S出现的概率就应该等于P(w1,w2,...,wn)，用条件概率的公式即得到共识①如下：
    $P(S)=P(w1,w2,...,wn)=P(w1)P(w2|w1)...P(wn|w1,w2,...,w(n-1))$
  - 考虑后一个词的出现概率只与前一个词相关
  - $P(S)=P(w1)P(w2|w1)...P(wn|w(n-1))$
- CBOW与Skip-gram模型
  - CBOW:通过上下文预测中间的词、Skip-gram:通过一个词预测上下文

Skip-gram

10000维
one-hot向量
(“机器”)

0		1	8	...	3	7		0.001	花朵
0		2	3	...	5	7		0.002	太阳
0		9	2	...	7	6		0.015	自然
...		...	...	...	...	...		...	...
1		1	2	...	3	4	1 2 ... 3 4	0.856	机械
...		...	...	...	...	...		...	...
0		7	3	...	5	9		0.003	天空
0		5	7	...	8	5		0.001	草地
0		6	3	...	5	4		0.001	风

1x10000维

10000x100维
词向量

1x100维

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$$

映射函数

1x10000维

fastText介绍

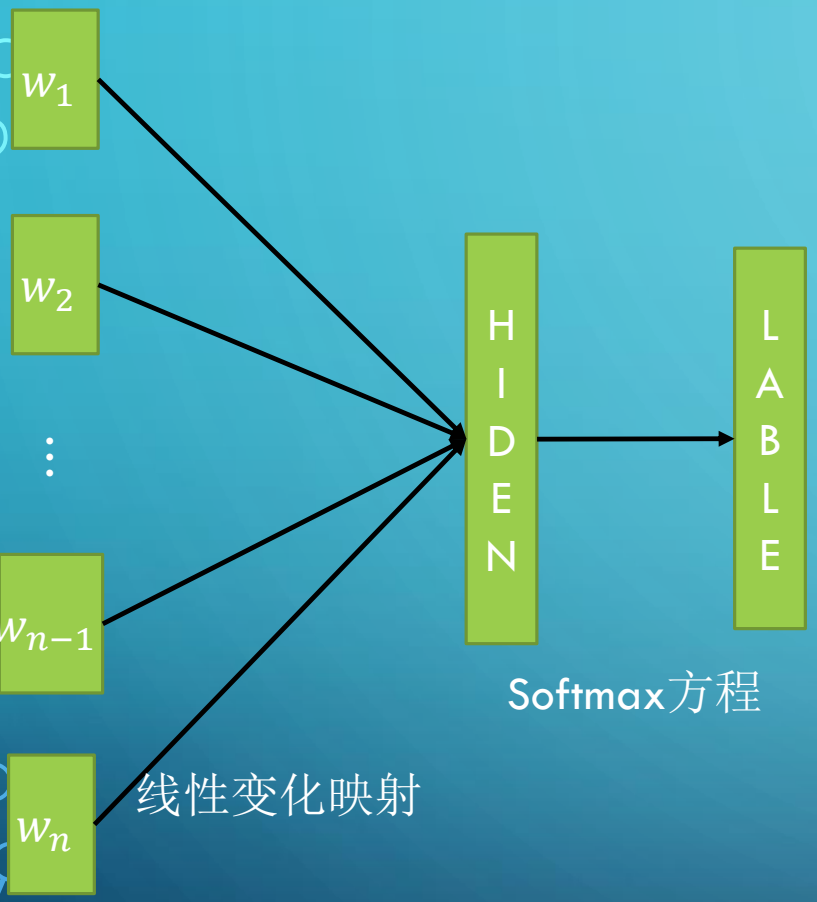
- 特点：fast
- 原因：一般情况下，文本学习并不需要过多隐藏层、过复杂的神经网

（一句话听几十遍与几遍，理解提升不高；一幅画临摹几十次与几次，差别很大）

fastText预备知识

- BoW(词袋模型)：忽略文本的语法语序，仅看作词汇集合，用一组无序词来表示一段文本，并假设文本中每个词的出现都是独立的。
  - 文本:”元芳 你 怎么 看”,”还能 怎么 看 趴 窗户 上 看”
  - 构建词典:{“元芳”:”1”, ”你”:”2”, “怎么”:”3”, “看”:”4”, “还能”:”5”, “趴”:”6”, “窗户”:”7”, “上 “:”8”}
  - 文本的向量表示:[1,1,1,1,0,0,0,0], [0,0,1,2,1,1,1,1],记录单词在文本中的频数
- 霍夫曼树（最优二叉数/带权路径最短的树）
  - 一棵二叉树中，n个叶子节点，Wi表示第i个叶子节点的权值，Li表示第i个叶子节点到根节点的路径长度，则该二叉树的带权路径长度：WPL=W1*L1+W2*L2+...Wn*Ln
  - 构造方法：①、给定n个权值看作只有根节点的n个二叉树集合HT;②、从HT中选出权值最小的两棵树，合并成新的二叉树，权值为两者之和;③、在HT中删去②中选出的两棵树，放入构造的新树;④、重复②③直至HT中只有一棵树，即为霍夫曼树

fastText模型架构



Softmax方程

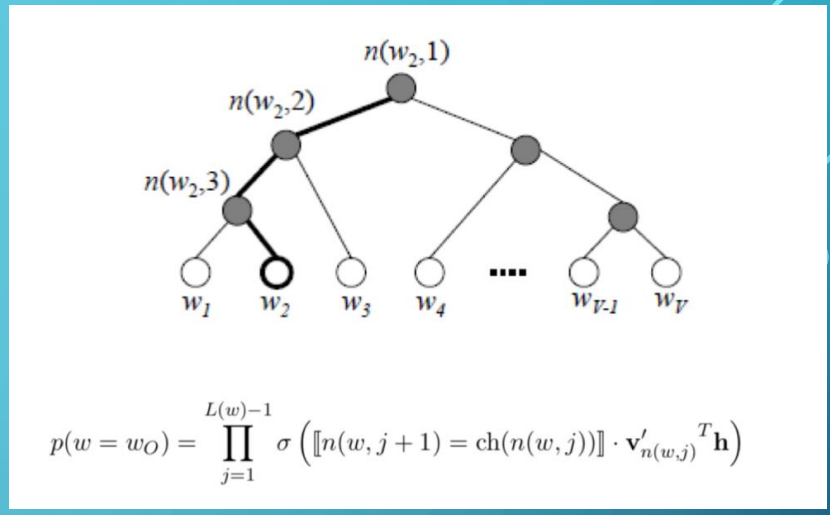
线性变化映射

词与词组成的特征向量 → 属于各标签的概率

(a1,a2,a3,a4…)

b1
b2
b3
b4
…

全局多分类，
计算代价O(V)，
V代表词汇数量



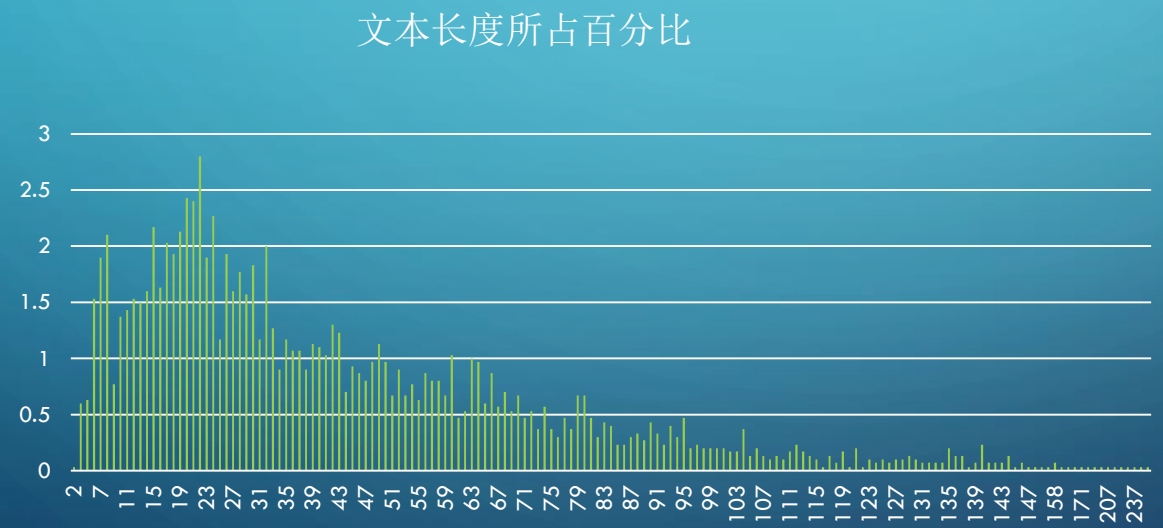
若干个二分类--词的路径可以预先统计（哈夫曼树是固定的）、中间每个逻辑回归单元对应向量可以学习和更新、隐藏层的输出
计算代价：O(log(V))

词作为基本单元—未登录词、低频词、拼写错误的词

subword n-gram:
google-><google>
Tri-gram:{<go,goo,oog,ogl,gle,le>}
…
一个词的向量由单词向量和字符级别的n-gram向量决定

分类任务及数据分析

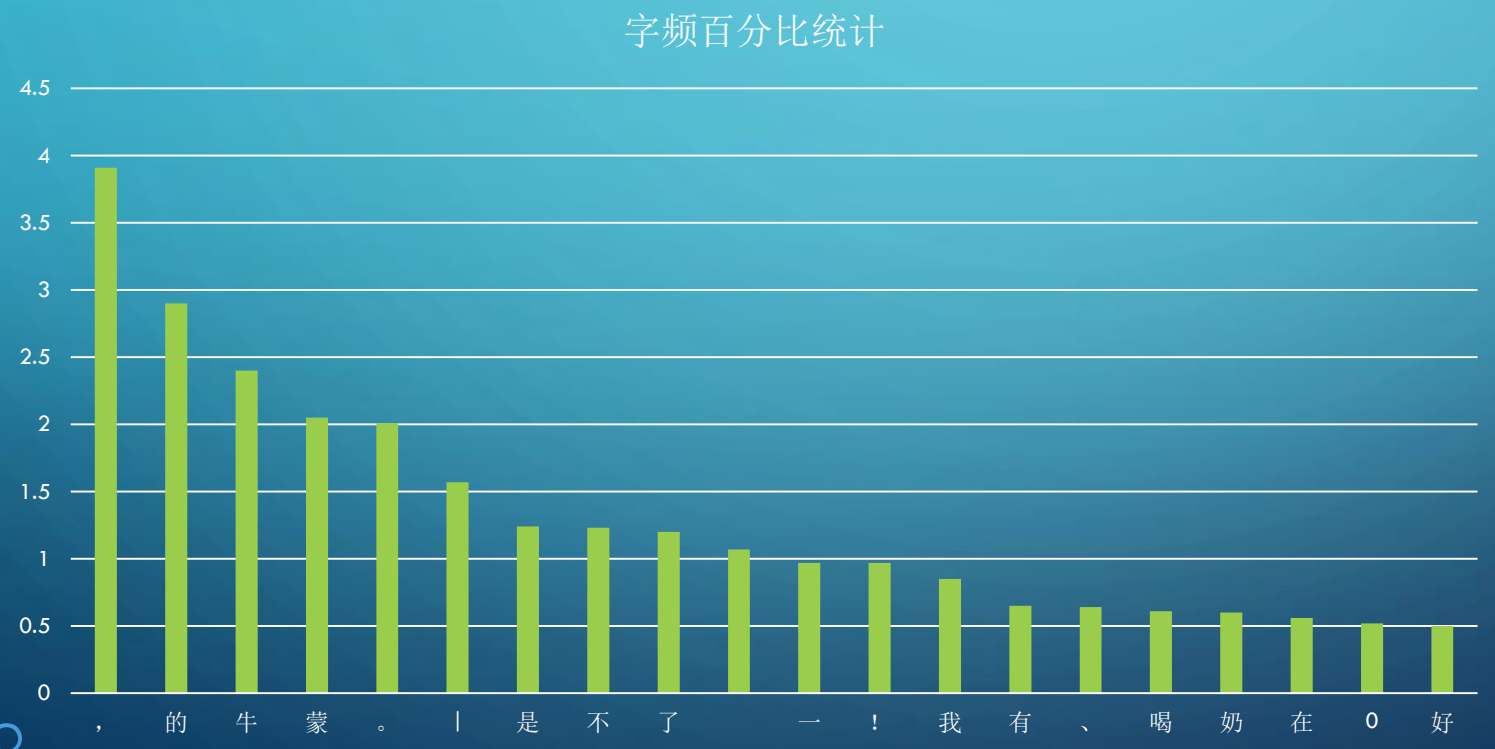
- 分类任务：给定文本，输出文本的情感倾向：正向，中立，负向
- 数据分析：
  - 数据量：3000条标注数据（三个类别各1000条）
  - 文本长度统计：共3000个句子，基本集中在20字左右，小于50个字的文本占68.27%，小于80个字的文本占87.73%



分类任务及数据分析

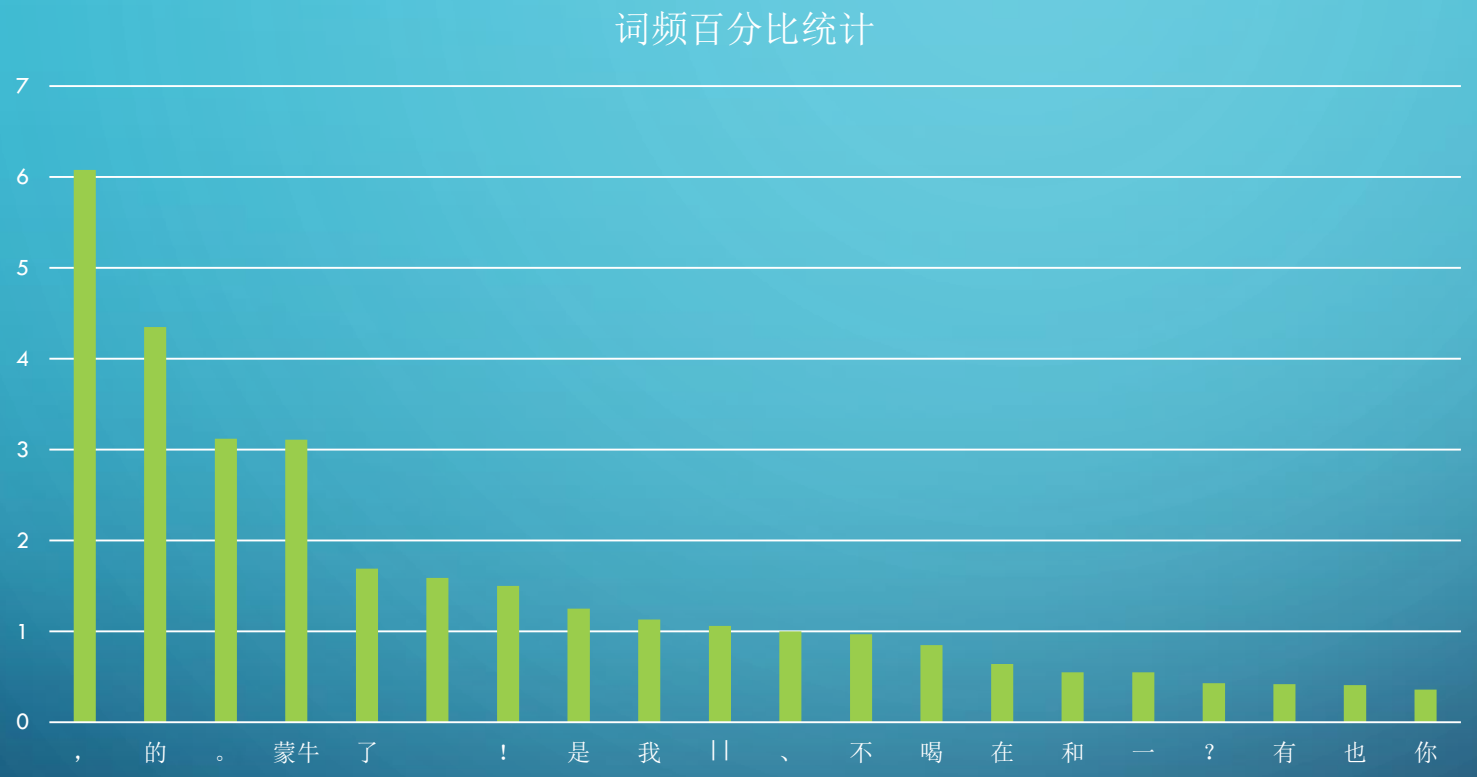
·字频统计：总共128465个字，3147个字类，排名前20的字占26.44%

标点符号：”，”、”。”、”|”等；”的”、”是”、”了”等。



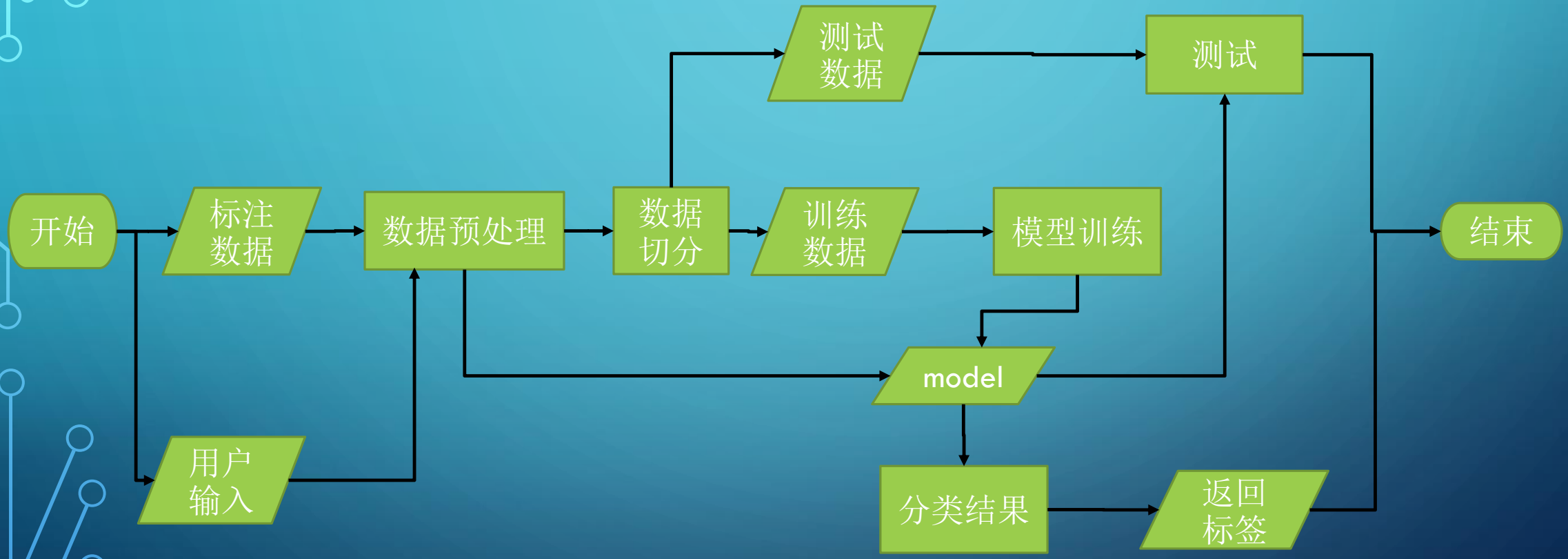
分类任务及数据分析

- 词频统计：总共82717个词，12381类词，排名前20的词占31.05%



分析结果：
文本中的无用信息：标点符号、虚词、领域词汇（“蒙牛”）

系统设计



数据预处理

- 词典数据
  - 停用词—stopwords.dic　　标点符号集—punctuations.dic
  - 积极词汇—positive_word.txt　　积极表情符—positive_emoticon.txt
  - 消极词汇—negative_word.txt　　消极表情符—negative_emoticon.txt
- ①识别表情符，进行替换
  - “:)”→”积极”　　“>_<”→”消极”
- ②识别标点符号，替换为空格
- ③对句子分词（先去除句子中的“蒙牛”）--结巴分词
- ④去除停用词
- ⑤识别情感词，为其打上特殊标记
  - “良心”→”# 良心 #”　　“黑心”→”* 黑心 *”
- ⑥标记label（1，0，-1）

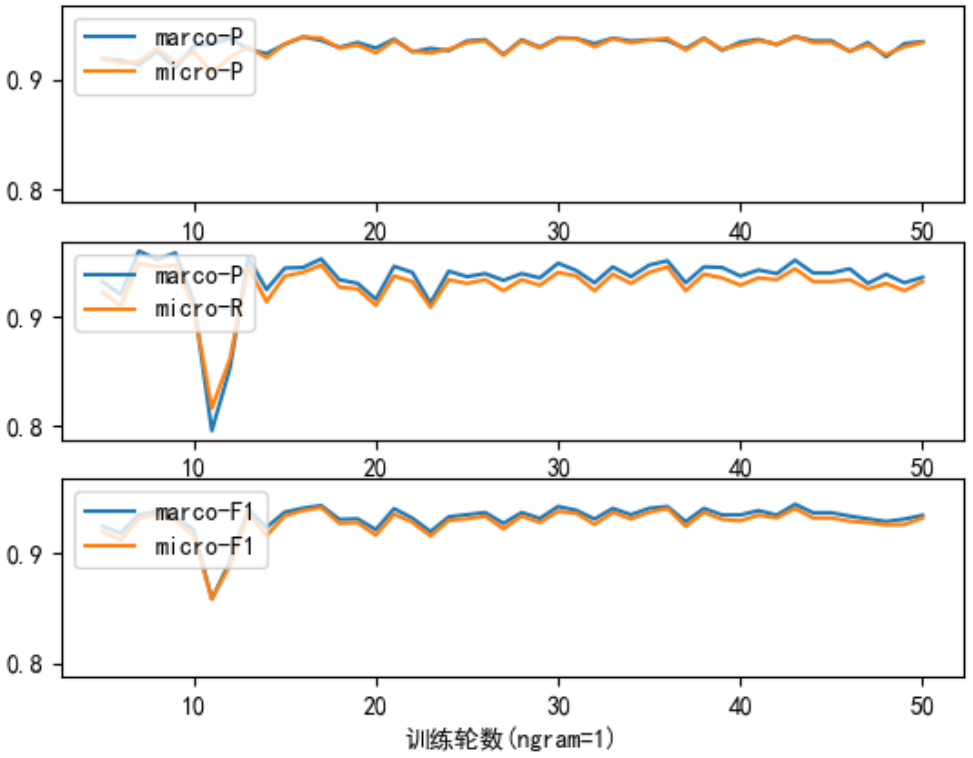
数据切分

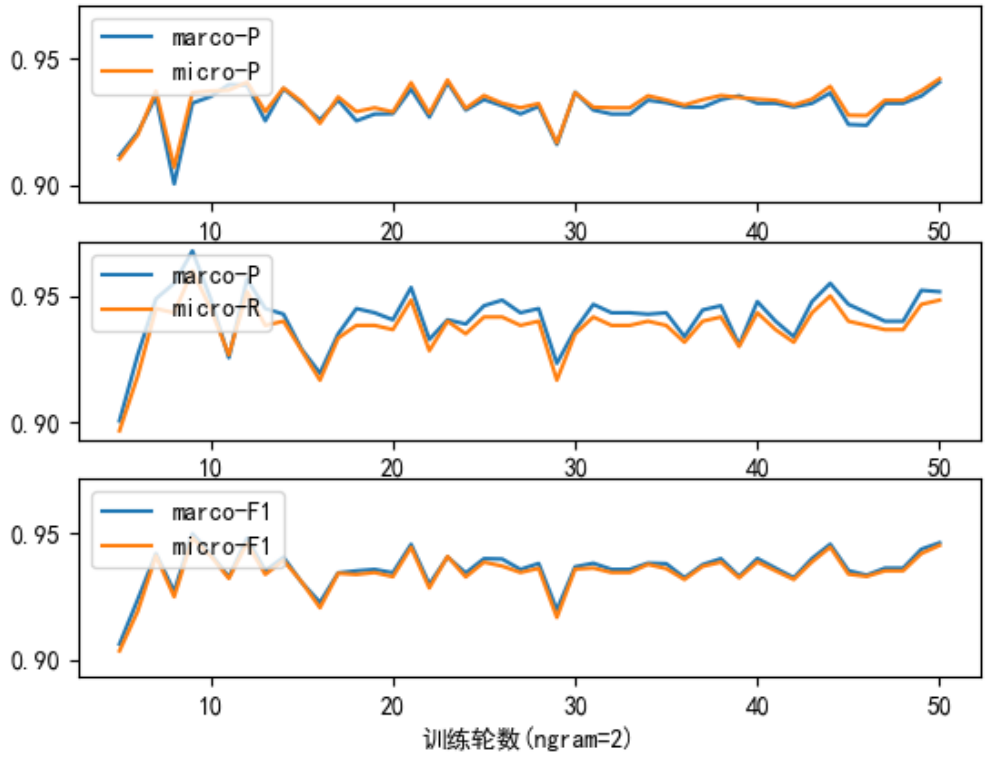
- 分组
  - 同类别标签，每10个分为一组
- 随机
  - 随机从10个中抽取一个作为测试集，其余作为训练集
- 统计
  - 训练集2700条，测试集300条

模型训练及实验结果

- 模型重要参数说明
  - -lr学习率：决定模型目标函数能否收敛到局部最小值以及何时收敛到最小值。
    - 较好的选取值：0.1-1.0
  - -epochs训练轮数：模型的训练次数[5-50]
  - -wordNgrams:n-grams的长度[1-5]
- 选择-lr 0.5,测试wordNgrams=1,2,分别运行5-50轮的情况

模型训练及实验结果





marco—宏平均　　micro—微平均　　F1=2*P*R/(P+R)

模型训练及实验结果

输入文本：

提交

提交文本：蒙牛真是好喝

计算结果：

正向------0.996094

负向------0.00195314

中立------1.95313e-08

提交文本：喝蒙牛，变傻牛

计算结果：

负向------0.998047

中立------1.95313e-08

正向------1.95313e-08

提交文本：桌子上有一层灰

计算结果：

中立------0.835938

正向------0.125

负向------0.0351563

提交文本：桌子上怎么有一层灰呢

计算结果：

负向------0.998047

中立------1.95313e-08

正向------1.95313e-08

Demo展示--HTTP://188.131.250.38/TEXT-EMOTION-COMPUTING

Github:HTTPS://GITHUB.COM/SANGYUHITER/TEXT-EMOTIONAL-COMPUTING

谢谢聆听！