点石文本分类大赛决赛

2018年01月

队长：骆金昌　队名：666

目录

比赛介绍与整体思路

比赛介绍

赛题提供一批网民真实的短文本评论数据，期望开发者通过建立模型分析出网民评论的的**正向、中立、负向**情感极性。

数据集 & 评价方法

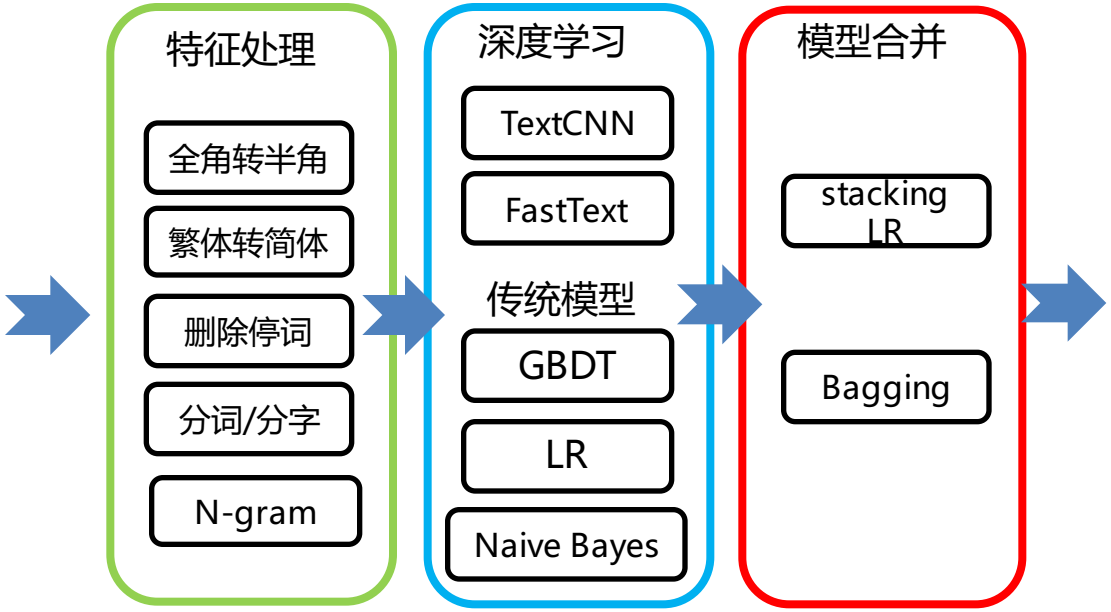
初赛设置A/B榜，提供**训练集19999条**、A榜提供测试集5000条，B榜提供测试集5000条。采用**加权F1**作为评价指标。

比赛排名

我们的方法取得成绩如下：

A榜第一名，B榜第三名

解决方案总架构及亮点



技术创新

1、深度学习模型与传统模型相结合，最好的解决方案合并了**100个模型**预测结果；

2、10-fold的数据分割，增强模型的泛化性与稳定性；

3、bagging训练方法，合理使用全量数据，提高整体泛化能力。

解决方案 | 文本预处理 & 特征提取

- **文本预处理**
  - 格式化
    - 全角转半角
    - 繁体转简体，nstool
  - 删除词或短语
    - 特殊字符，如：日本字符
    - 停词，如：的/是
    - 短语，如：回复XXX

- **文本特征提取**
  - 对句子进行：分词/分字
    - 分词：今天/我/很/开心
    - 分字：今/天/我/很/开/心

原句子：今天我很开心

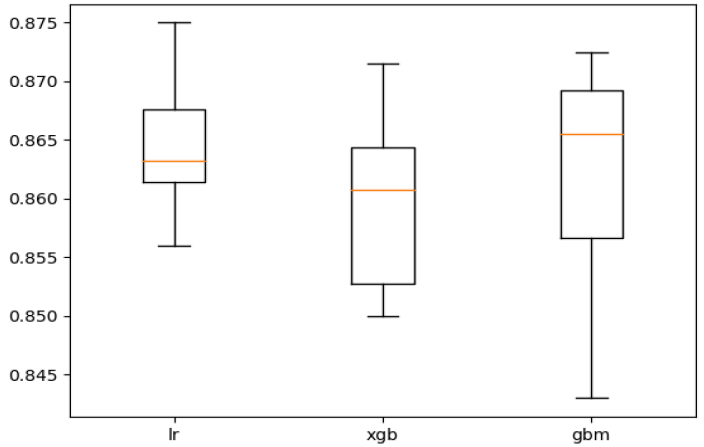
  - N-gram
    - n=1: 今/天/我/很/开/心
    - n=2: 今天/天我/我很/很开/开心
    - n=3: 今天我/天我很/我很开/很开心

解决方案 | 训练集分析及K-fold分割

超参选择：交叉验证

- 通过交叉验证方法确定超参选择
- 训练集分为K份，每次留出一个作为Valid集，剩下作为Train集，如上图
- 分别在不同fold中训练，得到该fold下Valid集的F1
- 取所有fold的**平均值作为该次超参数的效果**

如何选择选择 K

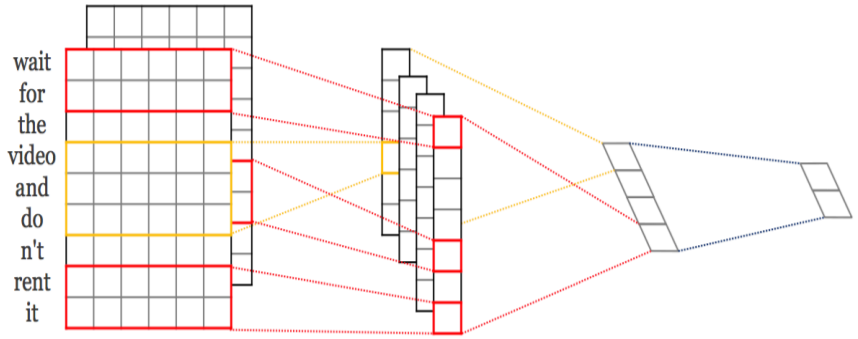


最好与最差的模型F1差距很大！

- 训练集非常小（2W），模型不稳定，如上图
- K=4或5时，Valid集效果好，但A榜效果不一定好，泛化性无法保证
- 当K越大时，两fold交集也越大，每fold利用更多数据，整体模型效果更稳定
- 综合权衡下，选择**K=10**

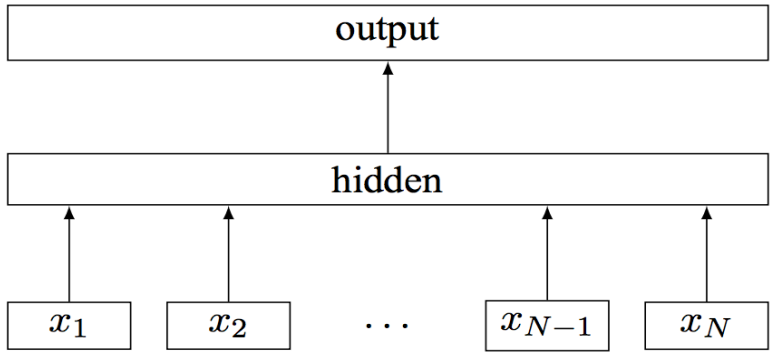
解决方案 | 深度学习模型

TextCNN



- 卷积层+Maxpooling层，提取句子词的局部相关性
- 通过调整卷积核大小捕捉比N-gram更丰富的信息
- 卷积核大小：1,2,3,4,5,6；词向量长度256，L2惩罚项

FastText

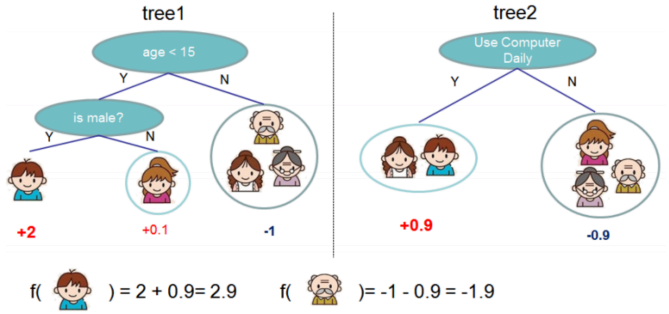


- 句子中所有的词向量进行平均，再接Softmax层分类
- 加入n-gram 特征的 trick 来捕获局部序列信息
- 文本进行分词或分字后作为特征输入
- 词向量长度100，迭代15次

解决方案 | 传统模型

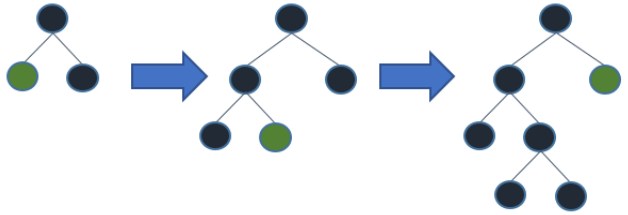
Xgboost

- 综合多棵树结果作预测，泛化性好，稀疏特征
- 诸多比赛中使用广泛，并且取得了不少好成绩



lightGBM

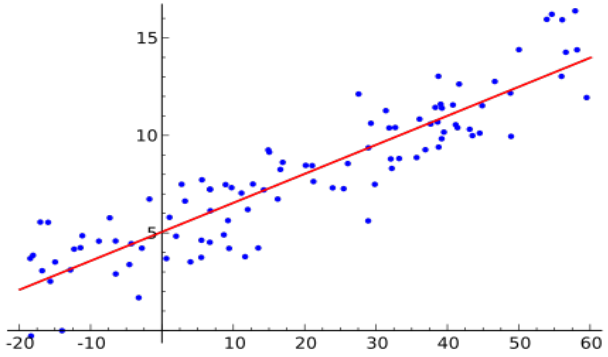
- 基于叶子(Leaf-wise)的树增长方式
- 更快速度，更少内存，更高的精度



Leaf-wise tree growth

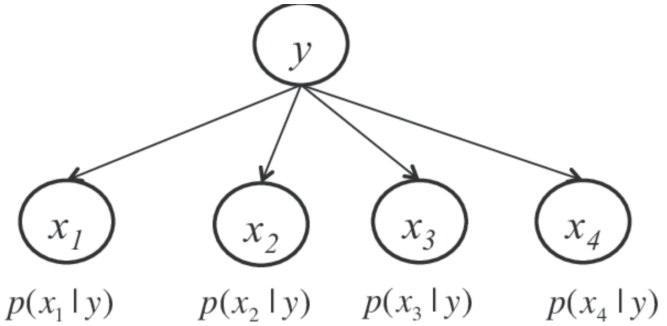
Logistic Regression

- 线性，简单高效、可解释性强
- 在工业界中广泛应用，如CTR预估

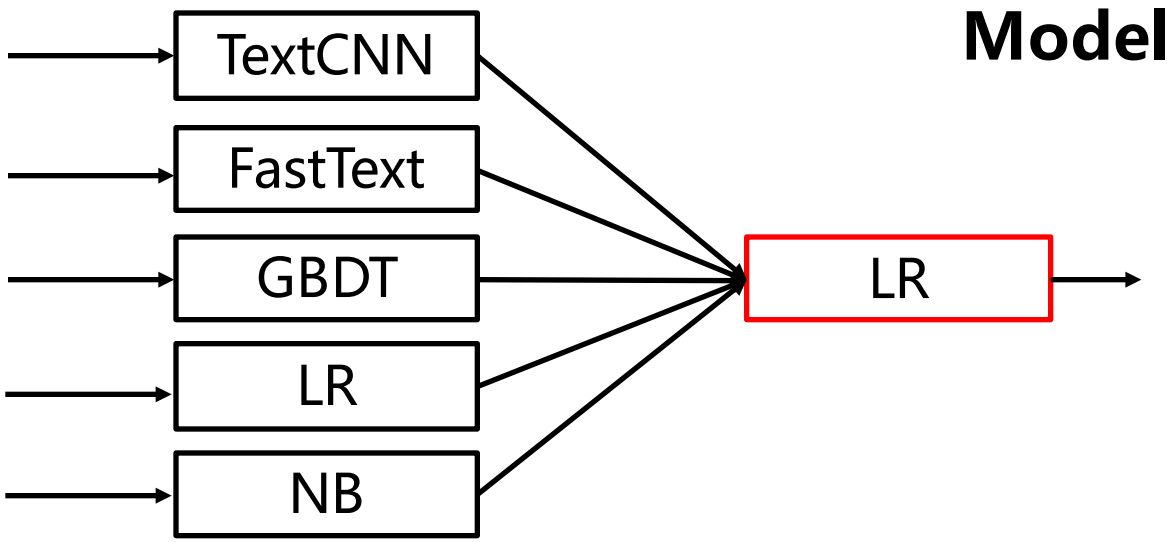


Naive Bayes

- 概率图模型、假设特征之间相互独立
- 特征分布： Bernoulli / Multinomial 分布

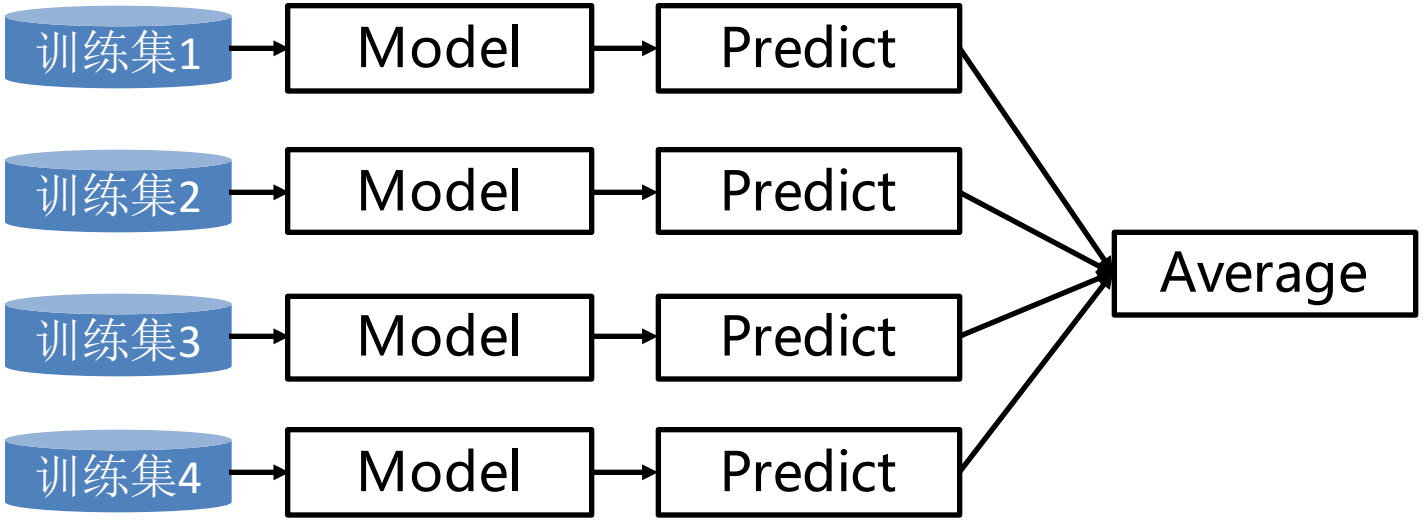


解决方案 | 模型合并



模型合并

- 相同的输入，独立训练基础模型GBDT、LR等
- 每个单独的模型预测三个极性的分布概率作为输出
- 综合所有单独模型的输出，构建新的训练集
- 重新在训练集上训练LR模型，合并所有模型的结果



类似Bagging的训练模式

- 训练集（2W）按10-fold分割，构造不同的训练集1、训练集2...
- 分别在训练集i上训练模型，得到测试集的预测结果
- 平均所有的测试集结果，提交到网站评估

解决方案 | 模型效果

Single Model			
Algorithm	**Feature**	**F1-Valid**	**F1-TestA**
BernoulliNB	N-gram	0.821	-
MultinomialNB	N-gram	0.803	-
LogisticRegression	N-gram	**0.864**	-
XGBoost	N-gram	0.860	0.803
lightGBM	N-gram	0.862	-
FastText	Char	**0.863**	0.792
	Word	0.843	-
TextCNN	Char	**0.863**	0.803
	Word	0.847	-

- 5个传统模型+4个深度学习模型
- 1个ensemble模型LR
- 10-fold取平均进行预测
- **总模型个数:(5 +4 + 1) * 10 = 100**

Combine Model			
Algorithm	**Ensemble**	**F1-Valid**	**F1-TestA**
FastText[Char] + FastText[Word]	Average	0.865	0.8057
FastText[Char] + FastText[Word] +XGBoost	LR	0.874	0.8143
Tradition Model + FastText[Char/Word] + TextCNN[Char]	LR	0.8770	0.8200
Tradition Model + FastText[Char/Word] + TextCNN[Char/Word]	LR	**0.8777**	**0.8210**

解决方案 | 排名

A榜第一名

排名	团队名	参赛者	f1-score	最优成绩提交日
1	666	jchluo123	0.8210	2018-01-16
2	成金	AaronLee22,dx,bert1018,lemondy9,stowho	0.8193	2018-01-16
3	天下第一	ms_xiaomao,搬砖的搬砖的搬砖的	0.8065	2018-01-16

B榜第三名

排名	团队名	参赛者	f1-score	最优成绩提交日
1	成金	AaronLee22,dx,bert1018,lemondy9,stowho	0.8060	2018-01-18
2	P90rushB	superDii,bleach92,mafng	0.7923	2018-01-19
3	666	jchluo123	0.7920	2018-01-18

比赛经验总结

什么路可能不work

- 特征层
  - 文本主题特征，如LDA、NMF，文本太短，主题模型效果不好
  - Ngram特征，N >=4 效果无提高，耗时无法接受
  - 语义向量特征+LR/SVM，效果无明显提升
  - TFIDF+Ngram，效果无明显提升
- 模型层
  - SVM跑2万样本，50万维Ngram特征，耗时无法接受
  - 随机森林，KNN效果差
  - LSTM模型在短文本中分类不稳定，需要提高模型稳定性
- ensemble层
  - 用GBDT作模型合并，质量无提高，容易过拟合，可能是因为特征维数太少
  - 一般来说，用LR作模型合并效果优于Average策略

比赛经验总结

什么方法可能work

- 特征层
  - 文本切分：单个模型，分字效果优于分词；组合模型，分字+分词效果更优
  - 参数选择：Ngram应该选择合适的N，这里为3最好
- 模型层
  - 模型比较　　：深度学习模型和传统模型同样重要，都可提高组合模型的效果
  - 高维稀疏特征：GBDT/LR可能更适合，高效、稳定、泛化性好
  - 模型实现　　：可以考虑多种实现，如GBDT，可以同时考虑XGBoost和lightGBM
  - 泛化性　　：尽量提高模型的泛化性与稳定性，如10-fold等
  - 探索　　：LSTM模型可能有效果，值得进一步探索
- ensemble层
  - 组合模型一般来优于单个模型，选择正确的组合算法，如LR/GBDT，组合效果可能非常惊人
  - 单个模型尽量要多样性，如树+线性+bayes+CNN等
  - stacking可以提高模型的泛化性

Q & A