# GBDT    适用于 linear

## 训练

1. 由 train1.data 生成 train1.model   (0002.model)
2. 用 train1.model 获取 train1.data 中每条记录的 leaf index 记作 train1.leaf
3. 将 leaf index 作为新特征 (index:1) 加入到 train1.data 末尾，并加上offset(原train1.data中最大index+1) 形成 train2.data
4. 用 train2.data 训练 train2.model

## 预测

1. 用 train1.model 得到 test1.data 的 leaf index (index offset 用 train1.data)
2. 将得到的 leaf index 作为新特征加入到 test1.data 末尾，形成 test2.data
3. 用 train2.model 预测 test2.data

# RNN

## 训练

1. 由 train1.data 生成 train1.model (同GBDT)
2. 由 train1.model 获取 train1.data 每条记录的 leaf index (同GBDT) 记作：train1.leaf
3. 将 train1.leaf 中的内容(leaf index序列，去掉tree index)看做分好词的文本, 用word2vec生成 train1.leaf.wordvec, train1.leaf.classid
4. 用 article2vec 将 train1.leaf 中的每篇文章（leaf index 序列）转换成 vector (sumvec or classid) 记作：train1.leaf.articlevec
5. 将 train1.leaf.articlevec 与 train1.data 合并，articlevec 的 Index offset 是原 train1.data 中最大的+1，生成 train2.data
6. 用 train2.data 训练 train2.model

## 预测

1. 用 train1.model 得到 test1.data 的 leaf index 记作 test1.leaf
2. 生成 test1.leaf.articlevec, 使用 train1.leaf.wordvec, train1.leaf.classid 做 ref
3. 生成 test2.data
4. 用 train2.model 预测 test2.data

## 在线预测样本

1. 设有新到样本sample1，用 service xgboost (train1.model) 得到 leaf index 记作 sample1.leaf
2. 用 service article 得到 sample1.leaf 的 article vector，(vecdict用train阶段的)，记作 sample.leaf.articlevec
3. Sample2 = sample1 + sample1.leaf.articlevec 注意indexoffset = max\_idx(train.data)+1
4. 用 service xgboost (train2.model) 预测 sample2 得到最终结果。

# 操作命令

get train1.leaf

* 启动xgboost alg server  
  GLOG\_logtostderr=1 ./xgboost\_svr.bin -model\_in train1.model -algname booster -algmgr localhost:9001 -port 10080
* BigRLab 命令端执行  
  service booster leaf train1.data out.data
* 处理中间结果，按返回id排序并去掉id  
  cat out.data | sort -k1,1n | awk '{$1=""; print $0}' > train1.leaf

article2vec

* 启动article alg server  
  GLOG\_logtostderr=1 ./demo -algname jieba -algmgr localhost:9001 -port 10080 -vec wordvec -vecdict data/train1.leaf.wordvec -idx data/text\_class.annIdx  
  (注：-idx虽然用不到，但必须指定一个合法的文件以通过参数检查)
* BigRLab 命令端执行  
  service jieba article2vector train1.leaf out.data
* 处理中间结果  
  cat out.data | sort -k1,1n | awk '{$1=""; print $0}' > train1.leaf.articlevec

合并数据见代码注释 merge\_new\_feature.cpp