官方评价标准：

Benchmark: [Benchmark Kernel | Kaggle](https://www.kaggle.com/code/dborkan/benchmark-kernel/notebook)

评价指标链接[Jigsaw Unintended Bias in Toxicity Classification | Kaggle](https://www.kaggle.com/competitions/jigsaw-unintended-bias-in-toxicity-classification/overview/evaluation)

用3个子指标产生一个综合指标来衡量。

子指标

1. Overall AUC 全局AUC：

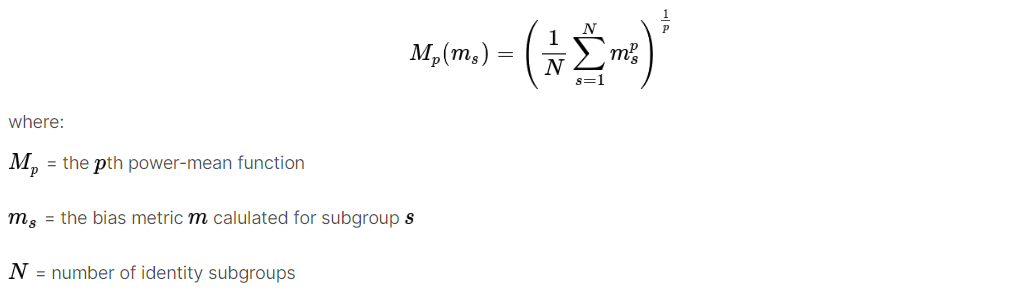
完整评估集的ROC-AUC。

二、Bias AUCs 偏差AUCc：

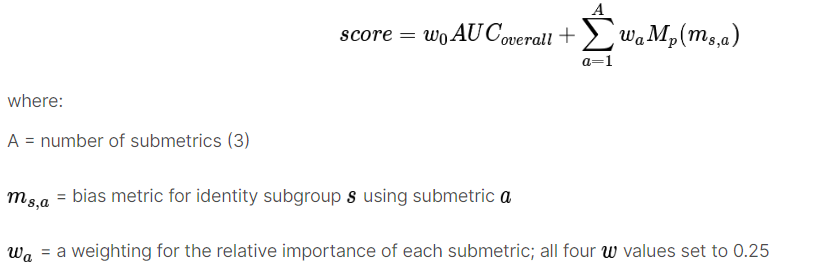
参考文献：[*Nuanced Metrics for Measuring Unintended Bias with Real Data in Text Classification*](https://arxiv.org/abs/1903.04561)，在测试集中按下面方法划分出三个子集，重新计算ROC-AUC

1. **Subgroup AUC 子群体AUC**:数据集限定仅为提及到身份的样本。如果结果较小说明模型在提及身份时鉴别毒性的能力很差。
2. **BPSN (Background Positive, Subgroup Negative) AUC ：数据集限定为提到身份的无毒样本 和 没有提及到身份的有毒样本。低值说明模型会将这两种样本混淆，模型预测值在提到身份的无毒样本中偏高。**
3. **BNSP (Background Negative, Subgroup Positive) AUC ：数据集限定为提到身份的毒性样本 和 没有提到身份的无毒样本。低值说明模型对提到身份的有毒样本的预测值偏低。**
4. Generalized Mean of Bias AUCs 偏差AUC的广义平均值：

利用Bias AUCs计算3个幂平均数，公式如下，按身份划分子群，计算指标的幂平均数，在本题中p取-5。



最终指标Score：上述4个子指标的加权计算，权重都设为0.25。



文本数据来源：[Civil Comments](https://medium.com/@aja_15265/saying-goodbye-to-civil-comments-41859d3a2b1d)  标签平台： [Figure Eight](https://www.figure-eight.com/)

选手其他数据集、预训练模型及标签：[official external data forum thread](https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification/discussion/87235) ：大多数是一些预训练模型的数据。

Benchmark Code解析：[Benchmark Kernel | Kaggle](https://www.kaggle.com/code/dborkan/benchmark-kernel/notebook)

1、文件输入:train.csv、test.csv和glove.6B.100d.txt文件。test是提交结果，不是真的测试集。glove.6B.100d.txt[来自GloVe: Global Vectors for Word Representation，一个用于无监督学习的词向量转换的语料库，在建模中使用到。](https://www.kaggle.com/datasets/rtatman/glove-global-vectors-for-word-representation" \t "_blank)

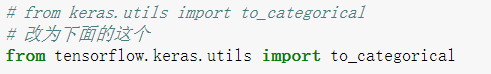
2、评论文本转成string类型。对于身份标签，按0.5阈值将连续型化为布尔型（模型评价中用用到）。  
3、把train按0.8和0.2划分训练集测试集。

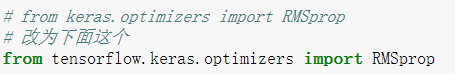
4、target代表毒性，也是标签，按0.5阈值转成bool型处理。

5、训练过程只涉及 文本内容和target 两个特征。

6、模型训练和预测（这里用的卷积神经网络）。

7、根据身份标签划分子集计算指标。

8、其他：代码导入包有两句需要修改



其他解答：总的来看LSTM、Bert用的比较多

1. [Wombat Inference Kernel | Kaggle](https://www.kaggle.com/code/iezepov/wombat-inference-kernel) 分数0.9471：最高分，代码极为复杂，无注释说明，可读性低，用了LSTM、Bert、GPT-CNN、fint-tune、base-uncased、cased、Wiki-ft和Wiki-cased进行加权计算预测。
2. [Fork of PRED lstm gpt2 bu12C45 Lu0C1 FTPe18 7c1c82 | Kaggle](https://www.kaggle.com/code/matsuik/fork-of-pred-lstm-gpt2-bu12c45-lu0c1-ftpe18-7c1c82) 分数0.94713：第二名，代码也极复杂，无注释说明，涉及到了毒性情感分量，但在建模的应用中只是把网络的层数设置为了毒性情感分量+1。用到了非公开的数据集。
3. [Simple LSTM | Kaggle](https://www.kaggle.com/code/thousandvoices/simple-lstm/script) 分数0.93424：投票最多，分数略低但只小了0.01，模型简单代码简洁，无注释说明，环境非jupyter应该是pycharm。运用的简单的lstm，输入文本内容，输出target及毒性情感分量（后者似乎是多余的？在预测部分完全没有用到这个输出结果）。
4. [Simple LSTM - PyTorch version | Kaggle](https://www.kaggle.com/code/bminixhofer/simple-lstm-pytorch-version) 分数0.9315：投票数第二，LSTM，有注释说明。Pytorch版前者。
5. [Toxic BERT plain vanila | Kaggle](https://www.kaggle.com/code/yuval6967/toxic-bert-plain-vanila) 分数未知：投票数第三，bert,有注释，代码可读性好，建模时仅应用到了文本内容和target，将target按0.5阈值化为布尔类型进行处理。