特征工程（下）—特征评估

原创 stephenDC 大数据与人工智能 2019-08-14

作者丨stephenDC

编辑丨Zandy

这是作者的第15篇文章

本文是特征工程系列的第3篇，也是最后一篇。

作者会在本文中结合自己在视频推荐方面的工作经验，着重从工程实现方面，讲述如何对特征进行评估的问题。下文中，我们首先会**厘清“特征评估”**的概念，然后讲述**特征评估的标准**，最后是**问题的反向排查。**

涉及到“**特征选择**”和“**特征表达**”的细节或背景，大家可以参阅该系列的前两篇文章，《特征工程（上）—特征选择》和《特征工程（中）-特征表达》。

厘清概念

什么是特征评估？

特征评估从概念上很容易跟特征选择纠缠到一起，因此非常有必要先厘清概念。在特征选择的过程中，我们需要对特征的每个维度进行评估，来选择出相对更重要的特征。然后，对于选择出的特征维度，我们会根据原始数据，对特征进行编码，进而得到特征。

本文所说的特征评估，**指的是对已经生成的特征的整体评估，发生在特征选择和特征编码之后，**因此不要跟特征选择过程中的对单个特征维度相对重要性的评估弄混了。

评估标准

分析前需要优先考虑哪些特征？

特征工程的**最终目的是提供给模型做预测**，因此只要特征在模型上表现的好就够了。这话一点儿没错，但倘若特征的表现不如人意呢？我们有没有办法提前发现问题，或者说如果最后不得不推倒重来，怎么找到改进的方向。所以，在最终的定量分析之前，还需要从**特征的覆盖率、特征维度、定性分析**等各个方面，对特征进行先行评估。

- **覆盖率**

覆盖率指的是，能成功生成出特征的视频和用户，占全体视频和用户的比例。特征提取和生成的方法不同，其覆盖率也自然不同。对视频的特征生成来说，可以基于内容，也可以基于用户行为。这里讲述并对比3种方法，分别以**关键词**(简称为“标签”)、“**ALS**”和“**Word2vec**”。

> **标签：视频标签是内容的体现，因此是基于内容生成特征的典型方式。**有了视频标签，就可以用One-hot或者TF-IDF的方式，进行特征编码，进而得到特征。因此，这种方法的覆盖率，取决于视频标签的覆盖率。

ALS：ALS是Alternating Least Square的首字母缩写，是Spark mllib最早实现的算法之一。ALS是求解矩阵分解的一种典型方法，将用户对视频的评分矩阵进行分解，分别得到用户和视频的特征。因此，这种方法存在冷启动问题，无法覆盖到尚无用户操作过的视频，也无法覆盖尚未有过操作行为的用户。

Word2vec：Word2vec是谷歌提出的词嵌入模型，可以将一个词嵌入到特定的特征空间之中，并维持词与词之间的语义关系。如果每个视频当成一个单词，将每个用户观看过的视频当成一个句子，则可以用Word2vec对视频编码，得到视频特征。所以，这种方法从原理上决定了，对没有被播放过或刚上线不久未被用户充分选择的视频，因为得到的特征会不准确，都无法覆盖。

- **特征维度**

在实际工程实现的时候，特征的维度是一个非常重要的考虑因素。因为**特征的维度，跟特征的表达能力、适用的模型以及计算的复杂度都有关系。**特征维度太低，显然特征的表达能力有限；特征维度太高，不仅会让计算量升级（**计算量跟模型复杂度有关**），还容易带来维度灾难的问题（可参看《机器学习中的维度灾难》）。因此，特征的维度要和计算框架以及数据规模相适应。

首先是特征维度要和计算框架的能力相匹配。作者曾经基于**Spark mllib**来做特征工程，后来发现**mllib**有一个致命的缺陷，就是只实现了数据分布式，而没有实现参数分布式。我们准备了2亿条数据，3000多维的稠密特征，结果还只能抽样使用一部分数据来训练模型。后来我们**改用了Spark on Angel 的Parameter Server框架**，才算解决了这个问题。（Angel Git地址：https://github.com/Angel-ML/angel/tree/master/spark-on-angel）

然后，特征维度也要和数据规模相匹配。我们用另一种方案，为视频生成了稀疏特征，加上用户特征维度近千万级。显然，2亿条数据和千万级特征的情况下，用深度学习的模型，肯定会过拟合的。

- **定性分析**

对可解释性特征，基于抽样的定性分析非常有用。比如，使用视频标签来生成特征，每个维度都可解释。我个人非常喜欢“诺兰”、“姜文”、“周星驰”等标签，不喜欢“恐怖片”的标签，而我的播放记录也基本匹配了我的个人爱好，那么如果最后得到的特征在“恐怖片”这个维度比“诺兰”这个维度的值还要高，那就不用灌入模型训练了，这肯定是有问题的。

基于抽样的定性分析，还可以稍微拓展到公司的同事，请他们帮忙验证各自的特征。如果有明显的问题，我们就可以及时修正；如果已经没有显著的问题，我们就可以将特征灌入模型训练，然后基于模型结果进行定量分析了。

- **定量分析**

特征灌入模型，并对结果进行定量分析，这非常接近模型评估的概念了。对于一个训练好的模型，重要的是评估其泛化能力。对分类问题，可以在独立的测试集上考察**准确率**、**召回率**、**F-Score**等一系列指标；对回归问题，也有常规的MAE、RMSE等评估指标。当然，最终的模型评估，免不了要上线进行A/B测试，毕竟线下和线上的环境会有差异，而且有些商业指标可能上线之后才能统计。

对特征的评估，跟上述对模型的评估类似，稍微不同的是，我们现阶段评估的重心是特征。在模型评估的时候，特征已经确定，重点是选模型及最优的参数；而在特征评估的阶段，特征本身的好坏也需要作为一个变量，也即是说，如果这组特征在各种模型和参数之下的表现都不近人意，那我们就需要对问题进行反向排查了。

问题反向排查

特征表现不好，如何找出问题所在？

如果特征表现不好，怎么样去查找问题所在，找到改进方向呢？我们先来分析一下，最终的定量分析指标，比如RMSE，是怎么得到。首先，需要先从数据源提取出特征。在这个过程中，数据源的质量、特征提取的方案，都会影响到最后得到的特征。然后，我们将特征灌入模型进行训练，这时模型的选型、模型的训练，都会影响到最终的分析指标。所以，当特征表现不够好时，我们就可以按照这个流程对问题进行反向排查了。

1 模型的问题

首先，我们需要排除模型的问题。大家都知道，机器学习是没有免费午餐的，没有任何一种模型可以在任何问题上表现都优于另一种模型。所以，我们要根据特征和数据量，选择合适的模型。

> **举例：**

FM会对一阶特征进行二阶交叉，这对标签特征很有意义，那对Word2vec这样的嵌入特征是否就不一定合适了呢？再比如，KNN在低维问题上表现非常出色，但对于高维的特征，由于维度灾难的问题，也是不适合的。

如果这组特征在各种模型下的表现都不够好，此时我们就需要去考虑特征本身可能的问题了。

2 特征的问题

在数据源确定的情况下，影响特征质量的因素主要是特征选择和特征编码。在特征选择方面，我们要考虑选择出的特征是否完备，冗余度如何等。在编码方案上，我们也要考虑现有的编码方式，是否能合理地刻画一个对象。

举例：

比如，作者先前很排斥对电影标签用TF-IDF的方式编码。因为周星驰比赵本山演了更多的电影，“周星驰”这个标签比“赵本山”IDF（逆向文本频率）项的值就要低很多；但周星驰的电影风格很凸出，而《三枪》和《一代宗师》却并没有因为赵本山的参演而增加多少相似性。但后来从全局考虑，作者还是接受了TF-IDF的方案，而“周星驰”这个标签的重要程度只能从视频标签权重的角度补回来了。再比如，对用户看过的视频，如何根据播放行为转化为分数？要不要考虑观看行为的时间衰减，如何衰减？这些都会影响到最终的特征。

如果特征的编码方案经过排查，并没有大的问题，但特征表现还是不够好，那就需要去排查数据源的问题了。。

3 数据源问题

还是以视频推荐为背景，基于标签的特征构建，依赖视频标签数据和用户行为数据。也就是说，如果标签不准，或者用户行为里混入了脏数据，特征工程的质量是可以想见的。

这个问题很难完全避免，但需要相关团队的大力支持。

比如，视频标签需要编辑团队辛苦的打标签工作；而用户行为日志，则需要BI团队辛苦的ETL工作。没有这些相关同事的工作，特征工程就是巧妇难为无米之炊了。

小结

本文在“特征选择”和“特征表达”的基础上，聊了一下特征评估的问题。至此，特征工程系列终于结束。由于作者的水平及在这方面的工作经验均非常有限，个中不足还请大家不吝赐教。

-end-

相关内容阅读

1.特征工程（中）-特征表达

2.特征工程（上）—特征选择

3.指数分布族

4.误差反向传播

5.极大似然估计、极大后验估计和贝叶斯估计

6. 稀疏核机（下）—稀疏性

7. 稀疏核机（中）—核方法

8.稀疏核机（上）—SVM回顾